(19)日本国特許庁（ＪＰ）　(12) 公開特許公報（Ａ）　(11)特許出願公開番号
特開2002－185788
（P2002－185788A）
(43)公開日　平成14年6月28日(2002.6.28)

(51)Int.Cl.⁷	識別記号	ＦＩ	テーマコード(参考)
Ｈ０４Ｎ　1/405		Ｇ０６Ｆ　3/12	Ｌ　２Ｃ２６２
Ｂ４１Ｊ　2/52		Ｇ０６Ｔ　5/00	２００Ａ　５Ｂ０２１
Ｇ０６Ｆ　3/12		Ｈ０４Ｎ　1/40	Ｂ　５Ｂ０５７
Ｇ０６Ｔ　5/00	２００	Ｂ４１Ｊ　3/00	Ａ　５Ｃ０７７

審査請求　未請求　請求項の数22　ＯＬ　（全 34 頁）

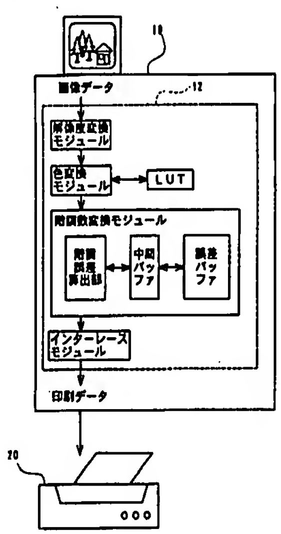
(21)出願番号　特願2001－238086(P2001－238086)
(22)出願日　平成13年8月6日(2001.8.6)
(31)優先権主張番号　特願2000－307896(P2000－307896)
(32)優先日　平成12年10月6日(2000.10.6)
(33)優先権主張国　日本（ＪＰ）

(71)出願人　000002369
セイコーエプソン株式会社
東京都新宿区西新宿2丁目4番1号
(72)発明者　角谷　繁明
長野県諏訪市大和三丁目3番5号　セイコーエプソン株式会社内
(74)代理人　110000028
特許業務法人　明成国際特許事務所

最終頁に続く

(54)【発明の名称】　画像処理装置、画像処理方法、印刷制御装置、および記録媒体

(57)【要約】
【課題】　画質の悪化を招くことなく、画像データを、ドット形成の有無によって表現された印刷データに迅速に変換する。
【解決手段】　ドット形成有無の判断により該判断画素で発生する階調誤差を、一旦、中間バッファに保持する。所定数の複数画素の判断を行った後、周辺の未判断画素に拡散される拡散誤差を、中間バッファに保持した階調誤差から算出して誤差バッファに拡散させる。こうすれば、階調誤差が発生する度に誤差バッファに拡散する場合に比べて、所定画素数分の階調誤差がまとめて拡散される分だけ、階調誤差を迅速に拡散することが可能となる。平均誤差最小法と数学的に同等な処理を、かかる方法を利用して実現することもできる。



1

【特許請求の範囲】
【請求項１】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理装置であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持する階調誤差保持手段と、
所定数の複数画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出し、該算出した拡散誤差を該未判断画素に対応付けて記憶する拡散誤差記憶手段と、
前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドット形成有無を判断するドット形成判断手段とを備える画像処理装置。
【請求項２】　前記階調誤差保持手段は、前記拡散誤差記憶手段に対して読み書きするよりも、前記階調誤差を迅速に読み書き可能な手段である請求項１記載の画像処理装置。
【請求項３】　請求項１記載の画像処理装置であって、
前記拡散誤差記憶手段は、
前記一時的に保持された階調誤差が前記未判断画素の各々に分配される分配誤差を、該階調誤差毎に算出して該各未判断画素に蓄積する分配誤差蓄積手段と、
前記所定数の複数画素についての前記分配誤差を全て蓄積する度に、前記蓄積されている全ての該分配誤差を、前記拡散誤差として前記未判断画素に対応付けて蓄積する拡散誤差蓄積手段とを備える画像処理装置。
【請求項４】　請求項１記載の画像処理装置であって、
前記階調誤差保持手段は、前記所定数の複数画素についての前記階調誤差を保持する手段であり、
前記拡散誤差記憶手段は、前記未判断画素の各々に拡散される前記拡散誤差を、前記保持されている所定数の階調誤差に基づいて算出し、該算出した全ての拡散誤差を該各未判断画素に対応付けて蓄積する手段である画像処理装置。
【請求項５】　請求項１記載の画像処理装置であって、
前記拡散誤差記憶手段は、
前記一時的に保持された階調誤差が前記未判断画素の各々に分配される分配誤差を、該階調誤差毎に算出して該各未判断画素に蓄積する分配誤差蓄積手段と、
前記所定数の複数画素分の分配誤差が蓄積された前記各未判断画素の分配誤差を、前記拡散誤差として該未判断画素に対応付けて記憶する分配誤差記憶手段とを備える画像処理装置。
【請求項６】　請求項１記載の画像処理装置であって、

2

前記階調誤差保持手段は、前記所定数の複数画素についての前記階調誤差を保持する手段であり、
前記拡散誤差記憶手段は、前記複数の未判断画素中にあって前記保持されている全ての階調誤差が拡散される特定画素への拡散誤差を、該保持されている複数の階調誤差から算出して、該特定画素に対応付けて記憶する手段である画像処理装置。
【請求項７】　請求項１記載の画像処理装置であって、
前記拡散誤差記憶手段は、
前記所定数の複数画素周辺の第１の所定領域内にある第１の未判断画素の各々については、該所定数の複数画素についての前記階調誤差に基づき前記拡散誤差を算出して、該第１の未判断画素に対応付けて記憶する第１の拡散誤差記憶手段と、
前記所定数の複数画素周辺にあって前記第１の所定領域内にはない第２の未判断画素については、前記階調誤差が算出される度に、該階調誤差に基づいて、該第２の未判断画素の各々への前記拡散誤差を算出して、該第２の未判断画素に対応付けて記憶する第２の拡散誤差記憶手段と、
を備える画像処理装置。
【請求項８】　請求項７記載の画像処理装置であって、
前記第２の拡散誤差記憶手段は、前記第１の所定領域を包含する第２の所定領域内の前記第２の未判断画素について、前記拡散誤差を算出して記憶する手段である画像処理装置。
【請求項９】　請求項８記載の画像処理装置であって、
前記ドット形成有無の判断に関わる条件に応じて、前記階調誤差を前記未判断画素に拡散させる範囲を、前記所定数の複数画素周辺の前記第１の所定領域と、前記第２の所定領域とに切り換える拡散範囲切換手段を備える画像処理装置。
【請求項１０】　請求項９記載の画像処理装置であって、
前記拡散範囲切換手段は、前記ドット形成有無の判断にかかわる条件として、前記所定数の複数画素の階調値と、該各画素についてのドット形成有無の判断結果とに基づいて、前記階調誤差を拡散させる範囲を切り換える手段である画像処理装置。
【請求項１１】　請求項１記載の画像処理装置であって、
前記拡散誤差記憶手段は、前記所定数の複数画素の階調誤差として、互いに隣接する画素の階調誤差に基づき前記拡散誤差を算出して、前記未判断画素に対応付けて記憶する手段である画像処理装置。
【請求項１２】　請求項１１記載の画像処理装置であって、
前記ドット形成判断手段は、前記未判断画素に対応付けて記憶された前記拡散誤差と、該未判断画素に隣接する画素からの前記拡散誤差とを考慮して、該未判断画素に

ついてのドット形成有無を判断する手段である画像処理装置。

【請求項１３】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理装置であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出し、該判断画素に対応付けて記憶する階調誤差記憶手段と、
前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持する階調誤差保持手段と、
前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データにおける前記着目画素の階調値に基づいて、ドット形成有無を判断するドット形成判断手段と、
前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出して、該検出した既判断画素の中で前記階調誤差が一時的に保持されていない既判断画素の階調誤差を前記階調誤差記憶手段から読み出し、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新して、該次着目画素の判断に供する階調誤差更新手段とを備える画像処理装置。

【請求項１４】　請求項１３記載の画像処理装置であって、
前記着目画素で生じた階調誤差を一時保持する一時保持手段を備え、
前記階調誤差更新手段は、前記一時保持されている階調誤差で、前記次着目画素についての判断に使用しない前記階調誤差を更新する手段である画像処理装置。

【請求項１５】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理方法であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持しておき、
所定数の複数画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出し、
前記算出した拡散誤差を前記未判断画素に対応付けて記憶し、
前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドット形成有無を判断する画像処理方法。

【請求項１６】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理方法であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して、該判断画素に対応付けて記憶し、
前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持しておき、
前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データにおける前記着目画素の階調値に基づいて、ドット形成有無を判断し、
前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出し、
前記検出した既判断画素の中で前記階調誤差が保持されていない既判断画素の階調誤差を読み出して、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新した後、該次着目画素の判断に供する画像処理方法。

【請求項１７】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の印刷データに変換し、印刷媒体上にインクドットを形成して画像を印刷する印刷部に対して、該印刷データを出力することで、該印刷部を制御する印刷制御装置であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持する階調誤差保持手段と、
所定数の画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出し、該算出した拡散誤差を該未判断画素に対応付けて記憶する拡散誤差記憶手段と、
前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドット形成有無を判断するドット形成判断手段と、
前記ドット形成有無の判断結果に基づいて前記画像データを前記印刷データに変換し、前記印刷部に出力する印刷データ出力手段とを備える印刷制御装置。

【請求項１８】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の印刷データに変換し、印刷媒体上にインクドットを形成して画像を印刷する印刷部に対して、該印刷データを出力することで、該印刷部を制御する印刷制御装置であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出し、該判断画素に対応付けて記憶する階調誤差記憶手段と、
前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持する階調誤差保持手段と、
前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データにおける前記着目画素の階調値に基づいて、ドット形成有無を判断するドット形成判断手段と、
前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出して、該検出した既判断画素の中で前記階調誤差が保持されていない既判断画素の階調誤差を前記階調誤差記憶手段から読み出し、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新して、該次着目画素の判断に供する階調誤差更新手段と前記ドット形成有無の判断結果に基づいて前記画像データを前記印刷データに変換し、前記印刷部に出力する印刷データ出力手段とを備える印刷制御装置。

【請求項１９】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する方法を実現するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持しておく機能と、
所定数の複数画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出する機能と、
前記算出した拡散誤差を前記未判断画素に対応付けて記憶する機能と、
前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドット形成有無を判断する機能とを実現するプログラムを記録した記録媒体。

【請求項２０】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する方法を実現するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して、該判断画素に対応付けて記憶する機能と、
前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持しておく機能と、
前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データに示された前記着目画素の階調値に基づいて、ドット形成有無を判断する機能と、
前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出する機能と、
前記検出した既判断画素の中で前記階調誤差が保持されていない既判断画素の階調誤差を読み出して、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新した後、該次着目画素の判断に供する機能とを実現するプログラムを記録した記録媒体。

【請求項２１】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する方法を、コンピュータを用いて実現するためのプログラムであって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持しておく機能と、
所定数の複数画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出する機能と、
前記算出した拡散誤差を前記未判断画素に対応付けて記憶する機能と、
前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドット形成有無を判断する機能とを実現するためのプログラム。

【請求項２２】　各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する方法を、コンピュータを用いて実現するためのプログラムであって、
前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して、該判断画素に対応付けて記憶する機能と、
前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎

に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持しておく機能と、

前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データに示された前記着目画素の階調値に基づいて、ドット形成有無を判断する機能と、

前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出する機能と、

前記検出した既判断画素の中で前記階調誤差が保持されていない既判断画素の階調誤差を読み出して、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新した後、該次着目画素の判断に供する機能とを実現するためのプログラム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】この発明は、画像を構成する複数の画素の階調値で表現された画像データを変換する技術に関し、詳しくは、該画像データを各画素についてのドットの形成有無による表現形式の画像データに変換する技術に関する。

【０００２】

【従来の技術】印刷媒体や液晶画面といった表示媒体上に、ドットを形成することによって画像を表現する画像表示装置は、各種画像機器の出力装置として広く使用されている。かかる画像表示装置は、局所的にはドットを形成するか否かのいずれかの状態しか表現し得ないが、画像の階調値に応じてドットの形成密度を適切に制御することによって、階調が連続的に変化する画像を表現することが可能となっている。

【０００３】これら画像表示装置において、画像の階調値に応じて適切な密度でドットが形成されるように、各画素についてドット形成の有無を判断するための代表的な手法としては、誤差拡散法と呼ばれる手法あるいは、これと数学的に等価な平均誤差最小法と呼ばれる手法などがある。

【０００４】誤差拡散法は、着目画素にドットを形成したこと、あるいはドットを形成しなかったことによって生じる階調表現の誤差を、着目画素周辺の未判断画素に拡散して記憶しておき、未判断画素についてのドット形成有無を判断するにあたっては、周辺画素から拡散されてきた誤差を解消するようにドット形成有無を判断する手法である。また、平均誤差最小法は、ドット形成有無の判断により生じた階調表現の誤差を周辺画素に拡散することなく着目画素に記憶しておき、その代わりに、未判断画素についてのドット形成有無を判断するに際しては、周辺画素に記憶されている誤差を読み出してこれらの誤差を打ち消すように、着目画素についてのドット形成有無を判断する手法である。これらいずれの手法においても、周辺画素で発生した階調表現の誤差を解消するようにドットの形成有無を判断しているので、画像の階調値に応じた適切な密度でドットを形成することができる。そのため、これらの手法を適用してドットの形成有無を判断すれば、高画質な画像を画像表示装置で表示することが可能となる。

【０００５】

【発明が解決しようとする課題】しかし、誤差拡散法には、着目画素で発生した階調表現の誤差を周辺の未判断画素に拡散しながらドットの形成有無を判断しているために、これら画素に誤差を拡散して記憶させる分だけ、ドット形成有無の判断に時間がかかってしまうという問題があった。また、平均誤差最小法についても同様に、周辺画素から階調表現の誤差を読み出しながらドットの形成有無を判断しているために、周辺画素から誤差を読み出す分だけ、ドット形成有無の判断に時間がかかるという問題があった。いずれの場合にも、ドット形成有無の判断に時間がかかれば、画像を迅速に表示することは困難となる。

【０００６】この発明は、従来技術における上述の課題を解決するためになされたものであり、誤差拡散法あるいは平均誤差最小法を適用した場合と同等の画質を維持しつつ、ドット形成有無の判断に要する時間を短縮化することによって、高画質の画像を迅速に表示可能な技術を提供することを目的とする。

【０００７】

【課題を解決するための手段およびその作用・効果】上述の課題の少なくとも一部を解決するため、本発明の第１の画像処理装置は、次の構成を採用した。すなわち、各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理装置であって、前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持する階調誤差保持手段と、所定数の複数画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出し、該算出した拡散誤差を該未判断画素に対応付けて記憶する拡散誤差記憶手段と、前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドット形成有無を判断するドット形成判断手段とを備えることを要旨とする。

【０００８】また、上記の第１の画像処理装置に対応する本発明の第１の画像処理方法は、各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理方法であって、前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持しておき、所定数の複数画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出し、前記算出した拡散誤差を前記未判断画素に対応付けて記憶し、前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドッット形成有無を判断することを要旨とする。

【０００９】かかる第１の画像処理装置あるいは画像処理方法においては、ドットの形成有無を判断することにより生じた前記階調誤差を一時的に保持しておき、所定数の複数画素についての階調誤差に基づいて、前記未判断画素への拡散誤差を算出してから記憶させる。こうすれば、所定数の複数画素の階調誤差を周辺の未判断画素にまとめて拡散して記憶させることができるので、各画素で生じた階調誤差を個別に拡散して記憶させる場合に比べて、迅速に記憶させることができる。その結果、ドット形成の有無の判断に要する時間が短縮化され、高画質の画像を迅速に表示させることが可能となる。

【００１０】かかる第１の画像処理装置においては、前記階調誤差記憶手段は、前記拡散誤差記憶手段に対して読み書きするよりも、前記階調誤差を迅速に読み書き可能な記憶手段としてもよい。

【００１１】前記所定画素数の階調誤差は、前記階調誤差記憶手段に一旦保持されてから前記拡散誤差記憶手段に記憶されるので、該階調誤差記憶手段は、該拡散誤差記憶手段と同様に頻繁に読み書きされる。従って、該階調誤差記憶手段を、該拡散誤差記憶手段に対して読み書きするよりも迅速に読み書き可能とすれば、該階調誤差が発生してから、最終的に拡散誤差として記憶されるまでの時間が短縮化され、延いては、ドット形成の有無の判断に要する時間を短縮化することができるので好適である。

【００１２】かかる第１の画像処理装置あるいは変換方法においては、前記拡散誤差を各未判断画素に対応付けて記憶するに際して、次のようにしても良い。すなわち、前記階調誤差が算出される度に、未判断画素の各々に分配される分配誤差を算出して、該算出した分配誤差を各未判断画素毎に蓄積していき、前記所定数の複数画素についての前記分配誤差を全て蓄積したら、各未判断画素毎に蓄積された全ての該分配誤差を、前記拡散誤差として前記未判断画素に対応付けて蓄積することとしてもよい。

【００１３】こうして階調誤差を求める度に、各未判断画素に分配誤差を蓄積しておけば、前記所定数の前記階調誤差を算出してから、前記拡散誤差を各未判断画素に対応付けて記憶させるまでの処理を迅速に行うことができる。所定数の複数画素分の階調誤差を１回拡散して記憶させただけでは、拡散すべき全ての階調誤差を記憶させることができない未判断画素についても、所定画素数分の拡散誤差を蓄積していくことにより、全ての階調誤差を拡散して記憶させることができる。従って、かかる方法を用いれば、拡散誤差を未判断画素に拡散して記憶させる処理を迅速に行うことができるので、ドット形成有無の判断に要する時間を短縮化することが可能となる。

【００１４】かかる第１の画像処理装置あるいは変換方法においては、拡散誤差を各未判断画素に対応付けて記憶するに際して、上述の方法に換えて次のようにしても良い。すなわち、所定数の複数画素についての階調誤差を保持しておき、保持されている所定数の階調誤差に基づいて各未判断画素に拡散される拡散誤差を算出して、該算出した全ての拡散誤差を各未判断画素に対応付けて蓄積することとしても良い。

【００１５】こうして所定数の複数の階調誤差を保持しておけば、各未判断画素へ拡散すべき拡散誤差を算出する処理が簡便となり、延いては、該所定画素数の階調誤差を周辺の未判断画素へまとめて拡散する処理が簡便となるので好適である。また、該未判断画素数に比べれば該所定画素数の方が少ないので、かかる方法を用いれば、階調誤差を算出する度に各未判断画素に分配誤差を蓄積する方法に比べて、記憶容量を節約することができて好ましい。

【００１６】前述した第１の画像処理装置あるいは変換方法においては、拡散誤差を各未判断画素に対応付けて記憶するに際して、次のようにしても良い。すなわち、前記階調誤差が算出される度に、未判断画素の各々に分配される分配誤差を算出して、該算出した分配誤差を各未判断画素毎に蓄積していき、前記所定数の複数画素分の分配誤差が蓄積された各未判断画素の分配誤差を、前記拡散誤差として前記未判断画素に対応付けて記憶することとしてもよい。

【００１７】このように、階調誤差を求める度に各未判断画素に分配誤差を蓄積し、所定画素数分の分配誤差が蓄積された未判断画素の分配誤差を、前記拡散誤差として記憶することとすれば、該所定画素数の拡散誤差を、各未判断画素に対応付けて記憶させる処理を迅速に行うことが可能となるので好ましい。

【００１８】かかる第１の画像処理装置あるいは変換方法においては、拡散誤差を未判断画素に対応付けて記憶するに際して、上述の方法に換えて次のようにしても良い。すなわち、所定数の複数画素についての階調誤差を保持しておき、複数の該未判断画素中にあって該所定数の階調誤差が全て拡散される特定画素への拡散誤差を、該保持している階調誤差から算出して、該特定画素に対応付けて記憶することとしてもよい。

【００１９】こうして所定画素数の階調誤差を保持しておき、前記特定画素への拡散誤差を算出して前記拡散誤差として記憶すれば、該拡散誤差を未判断画素へ拡散し

て記憶させる処理が簡便となる。その結果、該所定画素数の階調誤差を拡散して記憶する処理が迅速となり、延いては画像を表示するために要する時間を、短縮化することが可能となるので好ましい。

【0020】前述した第1の画像処理装置あるいは変換方法においては、拡散誤差を未判断画素に対応付けて記憶するに際して、次のようにしても良い。すなわち、前記所定数の複数画素周辺の第1の所定領域内にある第1の未判断画素の各々については、該所定数の複数画素についての前記階調誤差に基づき該第1の未判断画素への前記拡散誤差を算出して、該第1の未判断画素に対応付けて記憶する。また、前記所定数の複数画素の周辺にあって前記第1の所定領域内には含まれない第2の未判断画素については、前記階調誤差を算出する度に、該階調誤差から該第2の各未判断画素の各々へ拡散される前記拡散誤差を算出して、該第2の未判断画素に対応付けて記憶する。

【0021】こうすれば、前記第1の未判断画素については、所定数の複数画素から分配される分配誤差を各未判断画素にまとめて拡散することができるので、全体としてドット形成の有無を判断するために要する時間が短縮化されて好ましい。

【0022】かかる画像処理装置においては、階調誤差を算出する度に拡散誤差が算出されて拡散される前記第2の未判断画素として、前記第1の所定領域を包含した第2の所定領域内にある未判断画素に、前記拡散誤差を算出して記憶することとしても良い。

【0023】所定数の複数画素で生じた階調誤差を広い範囲に拡散させようとすれば、階調誤差の拡散範囲には、それだけ多くの未拡散画素が含まれることになるので、所定画素数の階調誤差をまとめて拡散して記憶させるための処理は複雑化する傾向にある。これに対して、階調誤差を拡散させる範囲を、前記所定数の複数画素の周辺の第1の所定領域と該第1の所定領域を包含する第2の所定領域とに分割し、該第1の所定領域にある第1の未判断画素については、所定数の複数画素についての階調誤差に基づいて各画素への拡散誤差を算出して記憶し、該第1の所定領域以外にある第2の未判断画素については階調誤差が算出される度に、該階調誤差からの拡散誤差を算出して、該第2の未判断画素の各々に対応付けて記憶させる。こうすれば、所定画素数の階調誤差をまとめて拡散すべき未判断画素数が減少するので処理を簡素化することが可能となって好適である。尚、第1の所定領域内にある第1の未判断画素に対して、所定数の複数画素についての階調誤差をまとめて拡散して記憶させるに際しては、上述した各種の方法を好適に適用することができるのはもちろんである。

【0024】かかる第1の画像処理装置あるいは変換方法においては、前記ドット形成有無の判断に関わる条件に応じて、前記階調誤差を周辺の画素に拡散させる範囲を、前記所定数の複数画素周辺の前記第1の所定領域と、該第1の所定領域を包含する第2の所定領域とに切り換えるようにしてもよい。

【0025】画質上の要請などから、ドット形成有無の判断に関わる条件に応じて、誤差を拡散させる範囲の広狭を切り換える技術が使用されることがある。このような場合に、階調誤差を拡散させる範囲を、所定数の複数画素周辺の第1の所定領域と、該第1の所定領域を包含する第2の所定領域とに切り換えるとともに、該第1の所定領域には含まれない遠方の未判断画素については階調誤差を直接拡散させることとすれば、所定画素数の階調誤差をまとめて拡散して記憶させる処理が簡素化するので好適である。

【0026】更には、前記所定数の複数画素の階調値と、該各画素についてのドット形成有無の判断結果とに基づいて、前記誤差を拡散させる範囲を、前記第1の所定領域と該第1の所定領域を包含する第2の所定領域とに切り換えることとしても良い。

【0027】例えば、表現しようとする画像データの階調値が小さな領域で、たまたまドットが形成された場合には、ドットが形成されたことによる誤差を広い範囲に拡散させれば、ドットの分散性を改善して画質を向上させることができる。あるいは、画像データの階調値が大きな領域で、たまたまドットが形成されなかった場合に、ドットが形成されなかったことによる誤差を広い範囲に拡散させれば、同様にして画質を向上させることができる。このような場合、前記第1の所定領域の未判断画素には前記所定数の複数画素からの拡散誤差をまとめて拡散するとともに、たまたまドットが形成された場合あるいは形成されなかった場合には、該第1の所定領域に含まれない遠方の未判断画素にも、階調誤差を算出する度に誤差を拡散する。こうすれば、遠方の未判断画素に誤差を拡散することは稀なので、実質的には処理速度をほとんど低下させることなく、ドット形成有無を判断する処理を簡素化することができるので好適である。

【0028】前述した第1の画像処理装置あるいは変換方法においては、前記所定数の複数画素を互いに隣接する画素として、該所定数の複数画素についての各階調誤差に基づき前記拡散誤差を算出して、前記未判断画素に対応付けて記憶することとしても良い。

【0029】階調誤差の発生した複数画素が互いに隣接している場合、互いに隣接していない場合に比べて、各画素で発生した階調誤差を拡散させる範囲の互いに重複する部分が広くなる。従って、各未判断画素への拡散誤差を、所定画素数分の階調誤差から算出して各画素に拡散した場合に、該所定数の複数画素が互いに隣接していれば、複数の階調誤差からの拡散誤差がまとめて拡散される未判断画素が多くなるので、それだけ、階調誤差を迅速に拡散して記憶させることが可能となって好適である。尚、かかる場合においても、所定画素数分の階調誤

差から算出した拡散誤差を未判断画素に拡散して記憶させるに際しては、上述した各種の方法を好適に適用可能なことは言うまでもない。

【0030】所定数の隣接した複数画素の階調誤差に基づいて拡散誤差を算出する上記の第1の画像処理装置あるいは変換方法においては、未判断画素のドット形成有無を判断するに際して、既に該未判断画素に対応付けて記憶されている拡散誤差に加えて、該未判断画素に隣接する画素からの前記拡散誤差を考慮して、ドット形成有無を判断することとしてもよい。

【0031】こうすれば、隣接する画素のドット形成有無が判断された未判断画素についてのドットの形成有無を判断する場合、隣接する画素から拡散されてくる拡散誤差については、該未判断画素に拡散させて記憶させる処理を行うことなくドットの形成有無を判断することができるので、判断を迅速に行うことが可能となって好適である。

【0032】また、前述した課題の少なくとも一部を解決するため、本発明の第2の画像処理装置は、次の構成を採用した。すなわち、各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理装置であって、前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出し、該判断画素に対応付けて記憶する階調誤差記憶手段と、前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持する階調誤差保持手段と、前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データにおける前記着目画素の階調値に基づいて、ドット形成有無を判断するドット形成判断手段と、前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出して、該検出した既判断画素の中で前記階調誤差が一時的に保持されていない既判断画素の階調誤差を前記階調誤差記憶手段から読み出し、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新して、該次着目画素の判断に供する階調誤差更新手段とを備えることを要旨とする。

【0033】上記の第2の画像処理装置に対応する本発明の第2の画像処理方法は、各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する画像処理方法であって、前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して、該判断画素に対応付けて記憶し、前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持しておき、前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データにおける前記着目画素の階調値に基づいて、ドット形成有無を判断し、前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出し、前記検出した既判断画素の中で前記階調誤差が保持されていない既判断画素の階調誤差を読み出して、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新した後、該次着目画素の判断に供することを要旨とする。

【0034】かかる第2の画像処理装置および画像処理方法においては、着目画素のドット形成判断に際しては、前記既判断画素に対応付けて記憶されている階調誤差を保持しておき、該保持されている階調誤差を考慮してドットの形成有無を判断する。次着目画素についてのドット形成有無を判断する際には、先ず初めに、該次着目画素についての既判断画素を検出し、次いで、該着目画素についての既判断画素との異同を調べる。そして、該着目画素のドット形成判断に使用しなかった既判断画素に記憶されている階調誤差を読み出し、読み出した階調誤差で、次着目画素の判断に使用しない階調誤差を更新した後、該次着目画素の判断に供する。

【0035】こうすれば、次着目画素のドット形成判断に際しては、該次着目画素についての判断には使用するが、着目画素についてのドット形成判断には使用しない階調誤差のみを読み出せばよい。その結果、ドット形成判断のために読み出さなければならない階調誤差数を減少させることができ、ドットの形成有無を判断するために要する時間を短縮化することが可能となる。

【0036】かかる第2の画像処理装置あるいは変換方法においては、前記着目画素で発生した階調誤差を一時保持しておき、該保持した階調誤差で、前記次着目画素についての判断に使用しない前記階調誤差を更新するようにしてもよい。

【0037】こうすれば、着目画素に対応付けて記憶されている階調誤差を読み出すことなく、次着目画素のドット形成判断を行うことができるので、迅速にドットの形成有無を判断することが可能となる。

【0038】また、印刷媒体上にインクドットを形成して画像を印刷する印刷部に対して、インクドットの形成を制御するための印刷データを出力することで、該印刷部を制御する印刷制御装置においては、本発明の上述した第1の画像処理装置あるいは第2の画像処理装置を好適に利用することができる。すなわち、上述した第1の画像処理装置あるいは第2の画像処理装置においては、各画素の階調値を示す画像データを受け取り、該画像データをドットの形成有無による画像データに迅速に変換することができる。このため、かかる第1の画像処理装

置あるいは第２の画像処理装置を、前記印刷制御装置に適用すれば、画像データを印刷データに迅速に変換することができる。こうして得られた印刷データを、前記印刷部に出力すれば、該印刷部では高画質な画像を迅速に印刷することが可能となるので好適である。

【００３９】また、本発明は、上述した第１の画像処理方法、あるいは第２の画像処理方法を実現するプログラムをコンピュータに読み込ませ、コンピュータを用いて実現することも可能である。従って、本発明は次のような記録媒体としての態様も含んでいる。すなわち、上述の第１の画像処理方法に対応する記録媒体は、各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する方法を実現するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して一時的に保持しておく機能と、所定数の複数画素についての前記一時的に保持された階調誤差に基づいて、該所定数の複数画素の周辺にあってドット形成有無の判断がなされていない未判断画素に拡散される拡散誤差を算出する機能と、前記算出した拡散誤差を前記未判断画素に対応付けて記憶する機能と、前記未判断画素に対応付けて記憶された前記拡散誤差を考慮しつつ、前記画像データにおける該未判断画素の階調値に基づいて、該未判断画素についてのドット形成有無を判断する機能とを実現するプログラムを記録していることを要旨とする。

【００４０】また、上述の第２の画像処理方法に対応する記録媒体は、各画素の階調値を示す画像データを受け取り、該画素毎に該階調値に基づきドット形成の有無を判断することによって、該画像データをドット形成の有無による表現形式の画像データに変換する方法を実現するプログラムを、コンピュータで読み取り可能に記録した記録媒体であって、前記ドット形成の有無を判断する度に該判断した画素で発生する階調誤差を、該ドット形成有無の判断結果に基づき算出して、該判断画素に対応付けて記憶する機能と、前記ドット形成の有無を判断しようとする着目画素の周辺にあって該ドット形成有無を判断済みの既判断画素毎に対応付けて記憶されている前記階調誤差を、該既判断画素毎に保持しておく機能と、前記既判断画素毎に保持されている階調誤差を考慮しつつ、前記画像データに示された前記着目画素の階調値に基づいて、ドット形成有無を判断する機能と、前記着目画素の次にドット形成有無を判断する次着目画素についての前記既判断画素を検出する機能と、前記検出した既判断画素の中で前記階調誤差が保持されていない既判断画素の階調誤差を読み出して、該読み出した階調誤差で該次着目画素についての判断に使用しない該階調誤差を更新した後、該次着目画素の判断に供する機能とを実現するプログラムを記録していることを要旨とする。

【００４１】これら記録媒体に記録されているプログラムをコンピュータに読み込ませ、該コンピュータを用いて上述の各種機能を実現すれば、画素毎の階調値を示す画像データを、ドット形成の有無による表現形式の画像データに迅速に変換することが可能となる。

【００４２】

【発明の実施の形態】本発明の作用・効果をより明確に説明するために、本発明の実施の形態を、次のような順序に従って以下に説明する。

Ａ．実施の形態：

Ｂ．第１実施例：

Ｂ－１．装置構成：

Ｂ－２．画像データ変換処理の概要：

Ｂ－３．第１実施例の階調数変換処理：

Ｂ－４．変形例：

Ｃ．第２実施例：

Ｃ－１．第２実施例において階調数変換処理の時間を短縮化する原理：

Ｃ－２．第２実施例の階調数変換処理：

Ｃ－３．変形例：

Ｄ．第３実施例：

Ｅ．第４実施例：

Ｅ－１．第４実施例において階調数変換処理の時間を短縮化する原理：

Ｅ－２．第４実施例の階調数変換処理：

【００４３】Ａ．実施の形態：図１を参照しながら、本発明の実施の形態について説明する。図１は、印刷システムを例にとって、本発明の実施の形態を説明するための説明図である。本印刷システムは、画像処理装置としてのコンピュータ１０と、カラープリンタ２０等から構成されている。コンピュータ１０は、デジタルカメラやカラースキャナなどの画像機器からカラー画像の階調画像データを受け取ると、該画像データを、カラープリンタ２０で印刷可能な各色ドットの形成有無により表現された印刷データに変換する。かかる画像データの変換は、プリンタドライバ１２と呼ばれる専用のプログラムを用いて行われる。尚、カラー画像の階調画像データは、各種アプリケーションプログラムを用いてコンピュータ１０で作成することもできる。

【００４４】プリンタドライバ１２は、解像度変換モジュール，色変換モジュール，階調数変換モジュール，インターレースモジュールといった複数のモジュールから構成されている。階調画像データをドットの形成有無による表現形式に変換する処理は、階調数変換モジュールで行われる。他の各モジュールで行われる処理については後述する。カラープリンタ２０は、これら各モジュールで変換された印刷データに基づいて、印刷媒体上に各色インクドットを形成することによってカラー画像を印刷する。

【００４５】本発明の印刷システムにおける階調数変換モジュールは、図示するように、階調誤差算出部と誤差バッファとの間に中間バッファが設けられた構成となっている。この中間バッファを活用することで、ドット形成有無の判断を迅速に行うことが可能となっている。

【００４６】すなわち、一般に、階調データをドット形成の有無によって表現しようとすると、おのずから、各画素には階調表現の誤差が発生する。前述した誤差拡散法あるいは平均誤差最小法と呼ばれる手法では、発生した階調誤差を、主記憶領域の一部に用意された大容量の誤差バッファに蓄積しておき、他の画素についてのドット形成有無を判断する際には、該画素に蓄積されている誤差を誤差バッファから読み出して、読み出した誤差の値を考慮しながらドット形成の有無を判断している。このことから明らかなように、ドット形成の有無を判断するためには、誤差バッファに対して頻繁にデータの読み書きを行う必要がある。誤差バッファには大容量のバッファが使用され、データの読み書きにある程度の時間を要するので、頻繁に読み書きすればその分だけ時間がかかってしまう。

【００４７】これに対して本発明の印刷システムにおける階調数変換モジュールでは、誤差バッファよりも小容量の中間バッファを活用することで、後ほど詳細に説明するように、誤差バッファに対して読み書きする頻度を減少させている。中間バッファは小容量の領域が繰り返して読み書きされるため、誤差バッファよりも高速に読み書きすることが容易である。例えば、主メモリとＣＰＵとの間に、主メモリよりも高速に読み書き可能なキャッシュメモリを有するコンピュータシステムにおいては、中間バッファからデータを読み出そうとすると、ほとんどの場合にキャッシュから読み出されると期待され、特別な配慮をしなくとも自然に中間バッファは誤差バッファよりも高速に読み書きすることが可能である。

【００４８】また、ＣＰＵのレジスタに余裕がある場合は、中間バッファの少なくとも一部を、キャッシュメモリよりも更に高速で読み書きが可能なレジスタに割り当てることが可能であり、更に高速化を図ることができる。中間バッファの一部をＣＰＵのレジスタに割り当てるには、例えばＣ言語ではregister宣言を用いることで、処理ソフトウェアのソースコードで明示的に指定することが可能である。更に、最適化能力の高いコンパイラの中には、余っているレジスタを使用頻度の高い変数に自動的に割り当てる機能を有するものがあり、このようなコンパイラを用いた場合には、明示的に指定せずとも、中間バッファの一部がＣＰＵのレジスタに割り当てられることになる。

【００４９】このように、繰り返し読み書きされる小容量の中間バッファは、通常のコンピュータシステムにおいては、特別な配慮を行わなくても高速に読み書き可能となっている。もちろん、ソフトウェアではなくハードウェアなどで処理する場合にも、小容量の中間バッファを用いた方が、高速に読み書き可能な構成を容易に実現することができる。

【００５０】データの読み書きにある程度の時間を要する誤差バッファへの読み書き頻度が減少すれば、その分だけドット形成有無の判断を迅速に行うことが可能となる。もちろん、中間バッファを用いていても、論理的には誤差拡散法あるいは平均誤差最小法と同等の処理を行うので、これら方法と同等の画質を維持することができる。誤差バッファへ読み書きする頻度を減少させるために中間バッファを活用する具体的な方法には、種々の態様が存在しており、これら各種の態様について、各種実施例を用いて以下に説明する。

【００５１】Ｂ．第１実施例：

Ｂ－１．装置構成：図２は、第１実施例の画像処理装置としてのコンピュータ１００の構成を示す説明図である。コンピュータ１００は、ＣＰＵ１０２を中心に、ＲＯＭ１０４やＲＡＭ１０６などを、バス１１６で互いに接続して構成された周知のコンピュータである。ＣＰＵ１０２は実際に処理を行う演算器と、処理中のデータを一時保持する複数のレジスタとから構成されている。レジスタに保持されているデータは、ＲＡＭ１０６に記憶されているデータよりも遥かに高速に処理することが可能である。もちろん、レジスタの代わりに、キャッシュメモリと呼ばれる特殊な記憶素子を活用しても良い。キャッシュメモリは、レジスタほどではないが、ＲＡＭ１０６よりも高速にデータを読み書きすることができる。また、キャッシュメモリを活用すれば、レジスタを使用する場合よりも多量のデータを扱うことが容易となる。

【００５２】コンピュータ１００には、フレキシブルディスク１２４やコンパクトディスク１２６のデータを読み込むためのディスクコントローラＤＤＣ１０９や、周辺機器とデータの授受を行うための周辺機器インターフェースＰ・Ｉ／Ｆ１０８、ＣＲＴ１１４を駆動するためのビデオインターフェースＶ・Ｉ／Ｆ１１２等が接続されている。Ｐ・Ｉ／Ｆ１０８には、後述するカラープリンタ２００や、ハードディスク１１８等が接続されている。また、デジタルカメラ１２０や、カラースキャナ１２２等をＰ・Ｉ／Ｆ１０８に接続すれば、デジタルカメラ１２０やカラースキャナ１２２で取り込んだ画像を印刷することも可能である。また、ネットワークインターフェースカードＮＩＣ１１０を装着すれば、コンピュータ１００を通信回線３００に接続して、通信回線に接続された記憶装置３１０に記憶されているデータを取得することもできる。

【００５３】図３は、第１実施例のカラープリンタ２００の概略構成を示す説明図である。カラープリンタ２００はシアン，マゼンタ，イエロ，ブラックの４色インクのドットを形成可能なインクジェットプリンタである。

もちろん、これら４色のインクに加えて、染料濃度の低いシアン（淡シアン）インクと染料濃度の低いマゼンタ（淡マゼンタ）インクとを含めた合計６色のインクドットを形成可能なインクジェットプリンタを用いることもできる。尚、以下では場合によって、シアンインク，マゼンタインク，イエロインク，ブラックインクのそれぞれを、Cインク，Mインク，Yインク，Kインクと略称するものとする。

【００５４】カラープリンタ２００は、図示するように、キャリッジ２４０に搭載された印字ヘッド２４１を駆動してインクの吐出およびドット形成を行う機構と、このキャリッジ２４０をキャリッジモータ２３０によってプラテン２３６の軸方向に往復動させる機構と、紙送りモータ２３５によって印刷用紙Ｐを搬送する機構と、ドットの形成やキャリッジ２４０の移動および印刷用紙の搬送を制御する制御回路２６０とから構成されている。

【００５５】キャリッジ２４０には、Kインクを収納するインクカートリッジ２４２と、Cインク，Mインク，Yインクの各種インクを収納するインクカートリッジ２４３とが装着されている。キャリッジ２４０にインクカートリッジ２４２，２４３を装着すると、カートリッジ内の各インクは図示しない導入管を通じて、印字ヘッド２４１の下面に設けられた各色毎のインク吐出用ヘッド２４４ないし２４７に供給される。各色毎のインク吐出用ヘッド２４４ないし２４７には、４８個のノズルNzが一定のノズルピッチkで配列されたノズル列が１組ずつ設けられている。

【００５６】制御回路２６０は、ＣＰＵ２６１とＲＯＭ２６２とＲＡＭ２６３等から構成されており、キャリッジモータ２３０と紙送りモータ２３５の動作を制御することによってキャリッジ２４０の主走査と副走査とを制御するとともに、コンピュータ１００から供給される印刷データに基づいて、各ノズルから適切なタイミングでインク滴を吐出する。こうして、制御回路２６０の制御の下、印刷媒体上の適切な位置に各色のインクドットを形成することによって、カラープリンタ２００はカラー画像を印刷することができる。

【００５７】尚、各色のインク吐出ヘッドからインク滴を吐出する方法には、種々の方法を適用することができる。すなわち、ピエゾ素子を用いてインクを吐出する方式や、インク通路に配置したヒータでインク通路内に泡（バブル）を発生させてインク滴を吐出する方法などを用いることができる。更には、インクを吐出する代わりに、熱転写などの現象を利用して印刷用紙上にインクドットを形成する方式や、静電気を利用して各色のトナー粉を印刷媒体上に付着させる方式のプリンタであっても構わない。

【００５８】また、インク吐出ヘッドから吐出するインク滴の大きさを制御したり、あるいは微細なインク滴を一度に複数吐出して、吐出するインク滴の数を制御するといった方法を用いて、印刷用紙上に形成されるインクドットの大きさを制御な可能なプリンタ、いわゆるバリアブルドットプリンタを用いることも可能である。

【００５９】Ｂ－２．画像データ変換処理の概要：図４は、第１実施例の画像処理装置としてのコンピュータ１００が、受け取った画像データに所定の画像処理を加えることにより、該画像データを印刷データに変換する処理の流れを示すフローチャートである。かかる処理は、コンピュータ１００のオペレーティングシステムがプリンタドライバを起動することによって開始される。以下、図４に従って、第１実施例の画像データ変換処理について簡単に説明する。

【００６０】プリンタドライバは、画像データ変換処理を開始すると、先ず初めに、変換すべきＲＧＢカラー画像データの読み込みを開始する（ステップＳ１００）。次いで、取り込んだ画像データの解像度を、カラープリンタ２００が印刷するための解像度に変換する（ステップＳ１０２）。カラー画像データの解像度が印刷解像度よりも低い場合は、線形補間を行うことで隣接画像データ間に新たなデータを生成し、逆に印刷解像度よりも高い場合は、一定の割合でデータを間引くことによって画像データの解像度を印刷解像度に変換する。

【００６１】こうして解像度を変換すると、カラー画像データの色変換処理を行う（ステップＳ１０４）。色変換処理とは、R，G，Bの階調値の組み合わせによって表現されているカラー画像データを、C，M，Y，Kなどのカラープリンタ２００で使用する各色の階調値の組み合わせによって表現された画像データに変換する処理である。色変換処理は、色変換テーブル（ＬＵＴ）と呼ばれる３次元の数表を参照することで迅速に行うことができる。

【００６２】色変換処理に続いて、階調数変換処理を開始する（ステップＳ１０６）。階調数変換処理とは次のような処理である。色変換処理によって変換された階調データは、各色毎に２５６階調幅を持つデータとして表現されている。これに対し、本実施例のカラープリンタ２００では、「ドットを形成する」，「ドットを形成しない」のいずれかの状態しか採り得ない。すなわち、本実施例のカラープリンタ２００は局所的には２階調しか表現し得ない。そこで、２５６階調を有する画像データを、カラープリンタ２００が表現可能な２階調で表現された画像データに変換する必要がある。このような階調数の変換を行う処理が階調数変換処理である。前述したように、本実施例においては中間バッファを活用することにより、ドットの形成有無の判断を迅速に行うことが可能となっている。階調数変換処理については、後ほど詳細に説明する。

【００６３】こうして階調数変換処理を終了したら、プリンタドライバはインターレース処理を開始する（ステップＳ１０８）。インターレース処理は、ドットの形成有無を表す形式に変換された画像データを、ドットの形成順序を考慮しながらカラープリンタ２００に転送すべき順序に並べ替える処理である。プリンタドライバは、インターレース処理を行って最終的に得られた画像データを、印刷データとしてカラープリンタ２００に出力する（ステップＳ１１０）。カラープリンタ２００は、印刷データに従って、各色のインクドットを印刷媒体上に形成する。その結果、画像データに対応したカラー画像が印刷媒体上に印刷される。

【００６４】以下では、第１実施例の階調数変換処理において、中間バッファを活用することにより、ドットの形成有無の判断を迅速に行う処理について説明する。

【００６５】Ｂ－３．第１実施例の階調数変換処理：中間バッファを活用してドット形成有無の判断に要する時間を短縮化する原理を説明するための準備として、いわゆる誤差拡散法においてドットの形成有無の判断を行う方法について簡単に説明する。

【００６６】図５は、誤差拡散法において、ドットの形成有無を判断した画素で発生した階調誤差を、周辺画素に拡散させながら、ドットの形成有無を判断していく様子を概念的に示した説明図である。尚、以下では、ドットの形成有無を判断しようと着目した画素を着目画素と呼ぶ。図５（ａ）に示すように、P00と符合した画素（着目画素）でドット形成有無を判断した結果として、該着目画素の画像データにおける階調値との間に階調誤差E00が発生したものとする。図中に斜線を施した部分はドット形成有無を判断済みの画素を示している。誤差拡散法では、階調誤差E00に所定の重み係数（誤差拡散係数）を乗算し、得られた値を、着目画素周辺の未判断画素に拡散する。尚、以下で使用する添え字は次の内容を意味するものとする。例えば、添え字「00」は着目画素を示し、添え字「01」は着目画素の右隣の画素を、添え字「0 -1」は左隣の画素を示す。添え字「10」は着目画素の真下の画素を、添え字「-1 0」は着目画素の真上の画素を示す。また、添え字「11」は添え字「10」と「01」とが組み合わされたものと考えて、着目画素の右斜め下の画素を示しているものとする。

【００６７】図６は、階調誤差を拡散する際に使用する誤差拡散係数の設定例を示す説明図である。尚、図６で斜線が付されている画素位置は着目画素の位置を示している。このような、着目画素から周辺画素への誤差拡散係数を表示したマトリックスは、誤差拡散マトリックスと呼ばれる。例えば、図６（ａ）の誤差拡散マトリックスでは、着目画素の右隣には誤差拡散係数K01の値として「１／４」が設定されている。従って、このような誤差拡散マトリックスを使用すると、着目画素で発生した階調誤差の１／４の誤差が右隣の画素に配分されることになる。同様に、着目画素の左下，真下，右下の各画素にも、着目画素で生じた階調誤差の１／４の誤差が配分される。誤差拡散マトリックスは図６に例示するものに限られず、誤差を拡散する範囲や誤差拡散係数などは種々の値を設定することが可能であり、実際の誤差拡散法では良好な画質が得られるように、適宜、適切な誤差拡散マトリックスが使用される。尚、説明の煩雑化を避けるために、以下の説明では、例示した誤差拡散マトリックスの中では最も拡散範囲の狭いマトリックス、すなわち、図６（ａ）の誤差拡散マトリックスを使用するものとして説明する。

【００６８】誤差拡散マトリックスとして図６（ａ）のマトリックスを使用するものとすれば、図５（ａ）に示すように着目画素P00で生じた階調誤差E00は、右隣の画素P01，左下の画素P1-1，真下の画素P10，右下の画素P11の合計４つの画素に、それぞれ階調誤差E00の１／４ずつ配分される。こうして着目画素周辺のそれぞれの画素に拡散された誤差（拡散誤差）は、画素毎に分離された状態で記憶しておく必要がある。そのため拡散誤差は、多数の画素についての拡散誤差を記憶することの可能な、大容量のＲＡＭ１０６（図２参照）に記憶される。

【００６９】画素P00についての階調誤差を周辺画素に拡散したら、今度は右隣の画素P01について、ドット形成有無の判断を開始する。図５（ｂ）は着目画素P01についてのドット形成有無を判断する様子を概念的に示す説明図である。ドット形成判断に際しては、先ず、周辺画素から着目画素P01に配分されて蓄積されている拡散誤差を読み出し、読み出した拡散誤差で着目画素P01の画像データを補正する。図５（ｂ）に示すように、着目画素P01には、ドット形成判断済みの周辺画素、すなわち画素P-10，画素P-11，画素P-12，画素P00の４つの画素から、前述の誤差拡散マトリックスに従って拡散されてきた誤差が蓄積されている。この拡散誤差をＲＡＭ１０６から読み出して、着目画素P01の画像データを補正し、得られた補正値を所定の閾値と比較することでドットの形成有無を判断する。判断の詳細については後述する。こうして着目画素P01についてドットの形成有無を判断すると、画素P01には新たな階調誤差E01が発生するので、この階調誤差を誤差拡散マトリックスに従って周辺画素に拡散する。このように誤差拡散法では、着目画素で階調誤差が発生する度に、その都度周辺にある複数の画素に拡散しながらドット形成有無を判断している。このため、ＲＡＭ１０６に対してデータを頻繁に読み書きする必要があるので、その分だけ、ドット形成有無の判断に要する時間も長くなる。

【００７０】これに対して本実施例の階調数変換処理においては、中間バッファを活用することにより、ドット形成有無の判断に要する時間を短縮化する。図７は、中間バッファを活用して、判断に要する時間を短縮化する原理を示す説明図である。本実施例では、ＣＰＵ１０２に内蔵されているレジスタを中間バッファとして使用す

る。前述したように、レジスタはＲＡＭ１０６に比べて高速に処理することが可能である。もちろん、レジスタではなく、キャッシュメモリを用いて実質的に同等の処理を行っても構わない。以下、図７を参照しながら、第１実施例においてドット形成有無の判断に要する時間を短縮化する原理について説明する。

【００７１】図７（ａ）は、着目画素Ｐ00についてドット形成有無を判断した様子を示している。該判断を行った結果として、着目画素には階調誤差Ｅ00が発生している。図７（ａ）の右側に表示した６つの矩形は、中間バッファとして使用される６つのレジスタを模式的に表示したものである。説明の便宜上、以下では、各レジスタにＲ01，Ｒ02，Ｒ1-1，Ｒ10，Ｒ11，Ｒ12と符合を付して区別する。第１実施例の階調数変換処理においては、着目画素Ｐ00で発生した階調誤差Ｅ00を、直接ＲＡＭ１０６に拡散するのではなく、一旦レジスタに蓄える。すなわち、レジスタＲ01の値を、着目画素の右隣の画素Ｐ01に配分すべき誤差で更新する。画素Ｐ01に配分すべき誤差は、誤差拡散マトリックスに従って、Ｋ01・Ｅ00で求めることができる。同様に、レジスタＲ1-1 の値を、着目画素Ｐ00の左下の画素Ｐ1-1に配分すべき誤差で更新し、レジスタＲ10の値を真下の画素Ｐ10に配分すべき誤差で、レジスタＲ11の値を右下の画素Ｐ11に配分すべき誤差で、それぞれ更新する。それぞれの誤差は誤差拡散マトリックスに従って、Ｋ1-1 ・Ｅ00，Ｋ10・Ｅ00，Ｋ11・Ｅ00で求めることができる。

【００７２】こうして周辺の４つの画素に配分すべきそれぞれの拡散誤差で、対応する４つのレジスタの値を更新したら、今度は画素Ｐ00の右隣の画素Ｐ01について、ドットの形成有無を判断する。画素Ｐ01の判断に際しては、新たな着目画素Ｐ01の拡散誤差をＲＡＭ１０６から読み出して、読み出した誤差とレジスタＲ01に記憶されている誤差とを加算した値で、着目画素Ｐ01の階調データを補正する。このように、着目画素Ｐ01の拡散誤差とレジスタＲ01の誤差とを加算した値で補正すれば、通常の誤差拡散法と、実質的にはまったく同じ処理を行うことができる。すなわち、通常の誤差拡散法では、図５を用いて説明したように、階調誤差が発生する度にこれを周辺の画素に拡散しておき、右隣の画素Ｐ01のドット形成有無を判断する際には、画素Ｐ01に配分されている拡散誤差を読み出して画素Ｐ01の階調データを補正し、補正した値に基づいてドットの形成有無を判断している。これに対して、本実施例においては、図７（ａ）に示すように、画素Ｐ01に配分される誤差がレジスタＲ01に記憶されているので、画素Ｐ01の拡散誤差とレジスタＲ01に記憶された誤差とを加算した値で、階調データを補正することで、通常の誤差拡散法と実質的に全く同等の処理を行うことができるのである。こうして新たな着目画素Ｐ01についてドットの形成有無を判断すると、着目画素Ｐ01には、新たな階調誤差Ｅ01が発生する（図７（ｂ）参照）。

【００７３】画素Ｐ01に発生した階調誤差Ｅ01も、画素Ｐ00の階調誤差Ｅ00と同様に、誤差拡散マトリックスに設定された割合で、それぞれのレジスタに拡散させる。図７（ｂ）を参照しながら具体的に説明する。先ず、画素Ｐ02は着目画素Ｐ01の右隣の画素であるから、対応するレジスタＲ02の値を、誤差拡散マトリックスに設定された誤差拡散係数Ｋ01と階調誤差Ｅ01とを乗算した値で更新する。画素Ｐ10は、着目画素Ｐ02の左下の画素であるから、対応するレジスタＲ10の値 には、誤差拡散係数Ｋ1-1 と階調誤差Ｅ01とを乗算した値Ｋ1-1 ・Ｅ01を加算する。ここで、図７（ｂ）に示すように、レジスタＲ10には、既に画素Ｐ00から拡散されてきた誤差Ｋ10・Ｅ00が記憶されているので、既に拡散されている誤差と新たに拡散されてきた誤差とをレジスタＲ02上で加算するのである。この結果、レジスタＲ02の値は、２つの画素から拡散されてきた誤差の和（Ｋ10・Ｅ00＋Ｋ1-1 ・Ｅ01）に更新される。以下、画素Ｐ11および画素Ｐ12についても同様に、対応するレジスタＲ11の値は拡散誤差（Ｋ11・Ｅ00＋Ｋ10・Ｅ01）で更新され、レジスタＲ12の値はＫ11・Ｅ01で更新される。

【００７４】以上のような処理を行うことにより、画素Ｐ00で発生した階調誤差Ｅ00と画素Ｐ01での階調誤差Ｅ01とが、それぞれのレジスタに拡散されたことになる。そこで、図７（ｃ）に示すように、各レジスタに記憶されている誤差を周辺画素に対応する誤差バッファ（具体的にはＲＡＭ１０６）に書き込んでいく。すなわち、画素Ｐ02の誤差バッファにはレジスタＲ02に記憶されている値Ｋ01・Ｅ01を加算し、画素Ｐ1-1 の誤差バッファにはレジスタＲ1-1 に記憶されている値Ｋ1-1 ・Ｅ00を、画素Ｐ10にはレジスタＲ10の値（Ｋ10・Ｅ00＋Ｋ1-1 ・Ｅ01）を、画素Ｐ11にはレジスタＲ11の値Ｋ11・Ｅ00＋Ｋ10・Ｅ01を、画素Ｐ12にはレジスタＲ12の値Ｋ11・Ｅ01を、それぞれ加算していく。こうすることにより、通常の誤差拡散法において画素Ｐ00および画素Ｐ01での誤差を拡散した状態（図５（ｃ）参照）と同様の誤差が、周辺の５つの画素の各誤差バッファに拡散されることになる。尚、画素Ｐ01については既にドット形成有無の判断を終了しているため、レジスタＲ01の値をＲＡＭ１０６に加算する必要はない。このように本実施例の方法では、２画素分のドット形成有無を判断して、５画素分の誤差バッファに誤差を拡散しているので、１画素分の判断をする度に２．５画素の割合で誤差バッファへの誤差拡散を行っていることになる。

【００７５】尚、以上では説明の便宜上、２値化を行った画素周辺の５つの画素に拡散される全ての誤差を、一旦レジスタに記憶し、２つの画素から拡散されてくる誤差をレジスタ上で加算してから誤差バッファに拡散するものとして説明した。もちろん、レジスタ上で誤差の加算を行わない画素、例えば画素Ｐ1-1 ，画素Ｐ02，画素Ｐ12については、誤差バッファに直接、誤差を拡散しても構わない。

【００７６】図５に示した通常の誤差拡散法では、１画素のドット形成判断を行う度に周辺の４画素に誤差を拡散していた。これに対して、本実施例の方法においては、未だ２値化していない同一の画素に、複数の画素から拡散される誤差を一旦レジスタ上で加算してからまとめて拡散することにより、１画素分の判断をする度に２．５画素分の誤差バッファに誤差を拡散するだけでよい。もちろん、通常の誤差拡散法に対して、誤差をレジスタへ配分する工程が追加されているが、レジスタの値の更新あるいはレジスタ上での加算といった処理はＲＡＭへ誤差を拡散させる処理に比べて遥かに高速に行うことができるので、全体としてドット形成有無の判断に要する時間を短縮化することが可能となる。尚、ここではレジスタに記憶するものとして説明するが、必ずしもレジスタに限られず、誤差バッファとして使用されている記憶手段よりも高速に読み書き可能な記憶手段でありさえすればよい。また、例えば、ＣＰＵとＲＡＭとのデータのやり取りを高速で行うために設けられているキャッシュメモリを活用することにより、実質的に同等な処理を行うものであっても構わない。本実施例で用いる中間バッファは、小容量のメモリ領域を各画素の処理に繰り返して使用するため、通常は設計者が明示的に指定しなくとも、コンパイラやＣＰＵ自体の機能によって自動的にレジスタやキャッシュメモリに割り当てられ、高速に読み書きすることが可能となる。

【００７７】尚、以上の説明では、２つの画素から拡散されてくる誤差をレジスタ上で直接、加算するものとしたが、それぞれの誤差を別々のレジスタに一旦記憶した後、別のレジスタ上で加算しても構わない。例えば、図７（ｂ）の画素Ｐ10に拡散される誤差を例にとって説明すると、先ずレジスタＲ10の値を画素Ｐ00からの誤差Ｋ10・Ｅ00で更新し、次いでレジスタＲ10上で、画素Ｐ01からの誤差Ｋ1-1 ・Ｅ01を加算している。このような処理に換えて、画素Ｐ00からの誤差と画素Ｐ01への誤差とをそれぞれレジスタＲa とレジスタＲb とに記憶しておき、２つのレジスタの値をレジスタＲ10上で加算しても構わないのはもちろんである。

【００７８】図８は、以上に説明した本実施例の階調数変換処理の流れを示すフローチャートである。この処理はコンピュータ１００のＣＰＵ１０２によって行われる。尚、前述したように、本実施例のカラープリンタは、Ｃ，Ｍ，Ｙ，Ｋの４色のインクドットを形成可能なプリンタであり、図８に示す階調数変換処理も各色毎に行っているが、以下では説明の煩雑化を避けるために、インクドットの色を特定せずに説明する。尚、上記の４色に加えて、ＬＣインク，ＬＭインクを加えて６色プリンタを使用してもよいのはもちろんである。

【００７９】また、前述したように本実施例のカラープリンタは、各色毎に大きさの異なるドットを形成可能なバリアブルドットプリンタとすることも可能である。バリアブルドットプリンタを使用する場合、例えば、大ドット，中ドット，小ドットの各種ドットを形成可能なバリアブルドットプリンタを用いる場合には、以下に説明する階調数変換処理は、各種大きさのドット毎に行われる。

【００８０】このように、使用するインクの色が増加したり、種々の大きさのドットが形成可能となるにつれて階調数変換処理を行う回数が増加するので、それだけ処理に要する時間も長くなる傾向がある。以下に説明する本実施例の階調数変換処理は迅速な処理が可能であるために、このような場合にも好適に適用することができる。

【００８１】本実施例の階調数変換処理を開始すると、先ず初めに、ドット形成有無を判断しようとしている２画素分の画像データＣd1，Ｃd2を読み込む（ステップＳ２００）。尚、ここでは便宜上、左側の画素を第１画素と呼び、右側の画素を第２画素と呼ぶことにする。画像データは、コンピュータ１００に内蔵されているＲＡＭ１０６に記憶されている。次いで、第１画素および第２画素のそれぞれに拡散されて蓄積されている拡散誤差Ｅd1，Ｅd2を誤差バッファから読み出す（ステップＳ２０２）。誤差バッファもＲＡＭ１０６上に設けられている。

【００８２】続いて、第１画素の画像データＣd1と第１画素の拡散誤差Ｅd1とを加算することにより、第１画素の補正データＣx1を算出する（ステップＳ２０４）。こうして得られた補正データＣx1と所定の閾値ｔｈ とを比較して（ステップＳ２０６）、補正データの方が大きければ第１画素にドットを形成すると判断して、第１画素についての判断結果を示す変数Ｃr1にドットを形成することを意味する値「１」書き込む（ステップＳ２０８）。そうでなければ第１画素にはドットを形成しないと判断して、変数Ｃr1にドットの形成しないことを意味する値「０」を書き込む（ステップＳ２１０）。

【００８３】こうして第１画素についてのドット形成有無を判断したら、該判断に伴って第１画素に発生する階調誤差を算出する（ステップＳ２１２）。第１画素で発生する階調誤差Ｅ1 は、第１画素の補正データＣx1から、ドットを形成することにより、あるいはドットを形成しないことにより第１画素で表現される階調値（以下では、このような階調値を結果値という）を減算することで求めることができる。

【００８４】次いで、第１画素で発生した階調誤差Ｅ1 を中間バッファに拡散する（ステップＳ２１４）。この処理は、図７（ａ）を用いて説明した処理に相当する処理である。すなわち、第１画素の階調誤差Ｅ1 に誤差拡散マトリックスに設定された誤差拡散係数を乗算して、周辺画素に拡散すべき拡散誤差を算出し、得られた拡散

誤差をそれぞれの中間バッファに記憶しておく。

【0085】第1画素で発生した誤差を中間バッファに拡散したら、第2画素についてのドット形成有無の判断を開始する。第2画素についての判断を行うためには、第2画素の補正データを算出する必要がある。そこで、予め読み出しておいた第2画素の画像データCd2と、第2画素の拡散誤差Ed2と、第1画素から第2画素に配分されてきた誤差とを加算して、第2画素の補正データCx2を算出する（ステップS216）。図7を用いて説明したように、第1画素から第2画素に拡散されてきた誤差は、予め定められた中間バッファに記憶されている。尚、図8に示す実施例では、第1画素の画像データおよび拡散誤差を読み出すと同時に、第2画素の画像データおよび拡散誤差も予め読み出しておくものとしたが、もちろん、第2画素については、補正データの算出に必要になってから、画像データと拡散誤差とを読み出すものとしても良い。第1画素および第2画素についての画像データと拡散誤差とを一度に読み出す場合には、メモリ上で連続した位置から読み出すといった方法を採用することによって、これらデータを迅速に読み出すことが可能になる。

【0086】尚、第2画素の画像データおよび拡散誤差については、ステップS200およびS202ではなく、ステップS216の直前で読み出すようにしても構わない。こうすれば、CPU102のレジスタ数に限りがある場合に、第2画素の画像データおよび拡散画素を読み出すまでの間は、レジスタを他の目的に使用することが可能となるので好適である。

【0087】ステップS216で第2画素についての補正データCd2を算出したら、第1画素の場合と同様に、所定の閾値thと比較することによってドットの形成有無を判断する（ステップS218）。補正データCd2の方が大きければ第2画素にドットを形成すると判断して、ドットを形成することを意味する値「1」を書き込み（ステップS220）、そうでなければドットを形成しないことを意味する値「0」を書き込む（ステップS222）。次いで、第2画素の階調誤差E2を算出する（ステップS224）。第2画素で発生する階調誤差E2も第1画素の階調誤差E1と同様にして、第2画素の補正データCx2から、第2画素についての結果値（画素についてのドットの形成有無を判断したことにより、その画素に表現される階調値）を減算することで求めることができる。

【0088】続いて、第2画素の階調誤差を中間バッファに拡散する処理を行う（ステップS226）。この処理は、図7（b）を用いて説明した処理に相当する処理である。すなわち、第2画素の階調誤差E2に誤差拡散マトリックスに設定された各画素の誤差拡散係数を乗算して、周辺画素に拡散すべき拡散誤差を算出し、得られた拡散誤差をそれぞれの中間バッファに記憶させる。ここで、中間バッファに第1画素から配分されてきた誤差が既に記憶されている場合には、既に記憶されている誤差に新たに配分した誤差を加算して記憶する。図7（b）に示した例では、レジスタR10およびレジスタR11の2つのレジスタについては既に配分されている誤差が存在するので、これに新たに配分した誤差を加算して記憶させる。この結果、5つのレジスタには第1画素および第2画素から、周辺の各画素に拡散すべき誤差が蓄積されることになる。そこで、続くステップS228の処理では、中間バッファに画素毎に記憶されている誤差を、RAM上の誤差バッファに加算し（図7（c）参照）、加算し終わったら中間バッファをリセットしておく。

【0089】次いで、全画素についてドット形成有無を判断したか否かを判断し（ステップS230）、未判断の画素が残っていればステップS200に戻って続く一連の処理を行う。こうして全画素について、ドット形成有無の判断を終了するまで、階調誤差を周辺画素に拡散しながら、2画素ずつドットの形成有無を判断し、全画素について判断を終了したら、本実施例の階調数変換処理を終了して、図4の画像データ変換処理に復帰する。

【0090】以上に説明した本実施例の階調数変換処理においては、2画素分のドット形成有無を判断する度に、5画素分の誤差バッファに誤差を拡散すればよい。通常の誤差拡散法を用いて同様の処理を行うためには、1画素の判断を行う度に4画素分の誤差バッファに誤差を拡散しなければならないことと比較すれば、本実施例の階調数変換処理を採用することで、誤差バッファへの誤差の拡散に要する時間を大きく短縮することができる。特に、階調数変換処理では、画像を構成する多数の画素についてドット形成有無を判断するとともに、発生した階調誤差を周辺画素に拡散していかなければならないので、誤差バッファへの誤差の拡散に要する時間は階調数変換処理に要する時間の中で比較的大きな割合を占めている。このことから、本実施例の階調数変換処理を採用して誤差の拡散に要する時間を短縮化すれば、階調数変換処理を含めた画像処理全体を迅速に行うことができ、延いては画像を迅速に印刷することが可能となる。

【0091】以上の説明では、説明の便宜上、2画素ずつドットの形成有無を判断するものとして説明した。すなわち、2つの画素で生じる階調誤差を中間バッファに蓄えておき、2つの画素から拡散されてきた誤差を中間バッファ上で加算し、まとめて誤差バッファに拡散する。もちろん、より多くの画素についてドット形成有無を判断し、発生した誤差を中間バッファ上で加算するようにしても良い。例えば、3画素ずつドットの形成有無を判断することとした場合は、図9（a）に示すように、ドット形成有無を判断する画素（図中で斜線を付して表示）の周辺にある6つの画素の誤差バッファに誤差を拡散すればよい。すなわち、判断を行う画素1つあたり、2画素分の誤差バッファに誤差を拡散すれば足りる。更に、6画素ずつドットの形成有無を判断した場合は、図9（b）に示すように、9画素分の誤差バッファに誤差を拡散すればよいので、判断画素1つあたりでは1．5画素分の誤差バッファに誤差を拡散すればよい。このように、一度にドット形成有無を判断する画素数を多くするほど、判断画素1画素あたりに拡散する誤差バッファの画素数が減少するので、階調数変換処理に要する時間を短縮化させることが可能になる。

【0092】B－4．変形例：また、以上に説明した第1実施例の階調数変換方法においては、第1画素および第2画素から周辺画素に拡散される拡散誤差を各階調誤差に基づいて算出し、周辺画素毎に蓄積しておくものとして説明したが、次のようにしても良い。すなわち、第1画素および第2画素で発生した階調誤差そのものを中間バッファに記憶しておき、これら記憶している階調誤差から周辺画素に拡散すべき拡散誤差を算出し、各画素に対応する誤差バッファに記憶させるようにしても良い。かかる方法は、中間バッファに記憶される値が蓄積中の拡散誤差であるか、階調誤差であるかの違いがあるものの、誤差バッファには同じ誤差が記憶されることから実質的に同等の処理を行うことができる。以下、このような第1実施例の変形例について説明する。

【0093】図10は、第1実施例の変形例における階調数変換方法の流れを示すフローチャートである。以下、図8に示した第1実施例の階調数変換処理と異なる部分を中心に、変形例の階調数変換方法について、図10を参照しながら簡単に説明する。

【0094】変形例においても第1実施例の階調数変換処理と同様に、処理を開始すると先ず初めに第1画素と第2画素の画像データCd1，Cd2を読み込む（ステップS300）。ここで、第1画素とはドット形成有無を判断しようとしている2つの画素の中の左側の画素を言い、第2画素とは右側の画素を言う。次いで、第1画素および第2画素のそれぞれに拡散されている拡散誤差Ed1，Ed2を誤差バッファから読み出す（ステップS302）。誤差バッファは第1実施例の場合と同様にRAM106上に設けられている。次いで、第1画素の画像データと拡散誤差とを加算して、第1画素の補正データCx1を算出する（ステップS304）。続いて、補正データCx1と所定の閾値thとを比較し（ステップS306）、補正データの方が大きければ第1画素にドットを形成すると判断して、第1画素についての判断結果を示す変数Cr1に、ドットを形成することを意味する値「1」を書き込む（ステップS308）。そうでなければ、第1画素にはドットを形成しないと判断して、ドット形成しないことを意味する値「0」を変数Cr1に書き込む（ステップS310）。こうして第1画素についてのドット形成の有無を判断したら、該判断に伴って第1画素に発生する階調誤差を算出する（ステップS312）。階調誤差は、第1実施例の場合と同様に、補正データCx1から結果値を減算することにより求めることができる。

【0095】第1実施例の変形例においては、こうして算出した第1画素の階調誤差E1を中間バッファに記憶する（ステップS314）。すなわち、第1実施例の階調数変換処理においては階調誤差を中間バッファに拡散させたが、変形例の階調数変換処理においては階調誤差を拡散させることなく、そのまま中間バッファに記憶するのである。

【0096】第1画素の階調誤差E1を中間バッファに記憶したら、第2画素についてのドット形成有無を判断するために、第2画素の補正データCx2を算出する（ステップS316）。すなわち、予め読み出しておいた第2画素の画像データCd2と拡散誤差Ed2と、第1画素から拡散されてくる誤差とを加算する。第1画素からの誤差は、中間バッファに記憶されている第1画素の階調誤差E1と誤差拡散係数とを乗算することで求められる。第1実施例の場合と同様に、誤差拡散係数の値は誤差拡散マトリックスに設定されている。こうして算出した第2画素の補正データCx2と所定の閾値thとを比較して（ステップS318）、補正データCx2の方が大きければドットを形成すると判断し、第2画素についての判断結果を示す変数Cr2にドットを形成することを意味する値「1」を書き込む（ステップS320）。そうでなければ、ドットは形成しないと判断して、変数Cr2にドットを形成しないことを意味する値「0」を書き込む（ステップS322）。次いで、第2画素についての階調誤差E2を算出し（ステップS324）、算出した階調誤差E2は、周辺画素に拡散させることなく、そのまま中間バッファに一旦記憶する（ステップS326）。このとき、第2画素の階調誤差E2は先に記憶しておいた第1画素の階調誤差E1とは別個に記憶しておく。第2画素の階調誤差E2は、第1画素の階調誤差E1と同様に、第2画素の補正データCx2から結果値を減算して求めることができる。

【0097】こうして、第1画素の階調誤差E1と第2画素の階調誤差E2とをそれぞれの中間バッファに記憶したら、2つの階調誤差をまとめて誤差バッファに拡散させる。この処理を、図7を利用して説明する。図7では、画素P00が第1画素に対応し、画素P01が第2画素に対応している。前述したように、第1画素および第2画素で発生した階調誤差は、これら画素の周辺にある5つの画素に所定の割合で拡散される。例えば、第1画素の左下にある画素P1-1には画素P00からの誤差K1-1・E00が拡散される。ここで、K1-1は、階調誤差を左下の画素へ拡散させる際に使用する誤差拡散係数であり、誤差拡散係数の値は誤差拡散マトリックスに設定されている。また、画素P10には、第1画素からの誤差K10・E00と、第2画素からの誤差K1-1・E01とを加算

した誤差K10・E00＋K1-1 ・E01が拡散される。他の３つの画素についても同様に、画素Ｐ11にはK11・E00＋K10・E01が、画素Ｐ12にはK11・E01が、画素Ｐ02にはK01・E01の値が、それぞれ拡散される。このように、周辺の各画素に拡散される誤差の値は、階調誤差E1 および階調誤差E2 が決まれば誤差拡散係数に基づいて算出することができる。そこで、ステップＳ３２８においては、中間バッファに記憶されている第１画素の階調誤差E1 と第２画素の階調誤差E2 とを用いて、第１画素と第２画素の周辺の各画素に拡散する誤差を算出し、各画素に対応する誤差バッファに加算していくのである。こうした処理が終了したら、次の画素の処理に使用するために、階調誤差E1 ，E2 を記憶している中間バッファをリセットしておく。

【００９８】次いで、全画素についてドット形成有無を判断した否かを判断し（ステップＳ３３０）、未判断の画素が残っていたらステップＳ３００に戻って続く一連の処理を行う。こうして全画素についてドット形成の有無を判断したら、第１実施例の変形例の階調数変換処理を終了して、図４の画像データ変換処理に復帰する。

【００９９】以上に説明した変形例の方法によっても、第１実施例の階調数変換処理と実質的に同等の処理を行うことができる。かかる変形例の方法によれば、第１実施例の方法よりも中間バッファの容量を節約することができる。

【０１００】Ｃ．第２実施例：

Ｃ－１．第２実施例において階調数変換処理の時間を短縮化する原理：以上に説明した第１実施例の階調数変換処理においては、ドット形成有無を判断して発生した階調誤差を、高速に読み書き可能なレジスタなどの中間バッファに拡散しておき、所定画素数のドット形成有無の判断を終了する度に、中間バッファに拡散されている全ての誤差を誤差バッファに加算していく。これに対して、以下に説明する第２実施例の階調数変換処理においては、ドット形成有無を判断する度に、１画素分ずつ誤差バッファに加算して行く。後述するように、第２実施例の方法によれば、第１実施例の方法よりも少ないレジスタ数で階調数変換処理を実現することができ、更に、レジスタなどの中間バッファから誤差バッファへの書き込み頻度も少なくすることができる。以下、かかる第２実施例について説明する。

【０１０１】初めに第２実施例の処理原理、すなわち、１画素のドット形成有無を判断する度に１画素分の拡散誤差を誤差バッファに加算して行くことによって通常の誤差拡散法と実質的に同等の処理を行う原理を、図１１を参照しながら説明する。図１１の斜線を付した部分は、ドット形成判断を行った画素の領域を示している。尚、以下でも、説明の便宜のために、第１実施例の場合と同様に誤差拡散マトリックスは図６（a）に示したマトリックスを使用するものとする。図１１（a）は、画素Ｐ00についてのドット形成有無の判断を行い、生じた階調誤差E00を周辺画素に拡散している様子を示している。

【０１０２】ここで、太い破線で囲って示した画素Ｐ11に着目すると、画素Ｐ11には、画素Ｐ00で生じた誤差E00に加えて、画素Ｐ00の右隣の画素Ｐ01で生じた誤差E01と、その更に右隣の画素Ｐ02で生じた誤差E02とが配分されている（図１１（b），（c）参照）。つまり１つの画素には、連続する３つの画素から続けて誤差が配分されていることになる。尚、連続する３つの画素から誤差が配分されるのは、図６（a）あるいは図６（e）に示す誤差拡散マトリックスを用いた場合で、他のマトリックス、例えば図６（b）や図６（f）のマトリックスを用いた場合は連続する５つの画素からの誤差が、図６（c）のマトリックスを用いた場合は連続する７つの画素からの誤差が、それぞれ続けて配分されることになる。このように、各画素には連続する画素から続けて誤差が配分されてくるので、配分されてくる誤差を画素毎に中間バッファ上で蓄積し、所定画素数の誤差を蓄積し終わったら、その画素についてはひとまず誤差の拡散が終了したと考えて、中間バッファに蓄積した誤差を誤差バッファに書き込んでいく。こうして、ひとまず誤差の蓄積が終了したと考えられる中間バッファの誤差を、対応する画素の誤差バッファに書き込んでいけば、同じ画素の誤差バッファに何度も誤差を書き込むする必要がなくなるので、誤差バッファへ読み書きする頻度を減らすことができる。

【０１０３】図１２は、上述した原理を、中間バッファを活用して実現している様子を概念的に示した説明図である。図１２（a）は、画素Ｐ00でドット形成有無を判断した状態を示している。図１２（a）の右側に示した４つの矩形は、それぞれ中間バッファとしてのレジスタを模式的に示したものである。以下では、各レジスタにR1 ，R2 ，R3 ，R4 と符号を付して識別する。着目画素Ｐ00で発生した階調誤差E00は、誤差拡散マトリックスに設定された誤差拡散係数が乗算されて、各レジスタに拡散される。尚、図示が煩雑化することを避けるため、図１２では、画素Ｐ00から左下の画素に拡散される誤差をE00__LDと略して表示する。誤差E00__LDの値は、画素Ｐ00の階調誤差E00に誤差拡散係数K1-1 を乗算することによって求めることができる。同様に、画素Ｐ00から真下の画素に拡散される誤差をE00__D と略して表示し、画素Ｐ00から右下の画素に拡散される誤差をE00　RDと、画素Ｐ00から右横の画素に拡散される誤差をE00__R と略して表示することとする。他の画素で発生した階調誤差についても同様に、例えば、画素Ｐ01から左下の画素に拡散される誤差はE01__LDと略して表示し、画素Ｐ01から真下の画素に拡散される誤差はE01　D と、右下の画素に拡散される誤差はE01__RDと、右横の画素に拡散される誤差はE01__R とそれぞれ略して表示する。また、各レジスタの右側に付した星印は、所定画素数の誤差を蓄積し終わったレジスタを示している。星印の意味する内容については後述する。

【０１０４】図１２（a）を参照して、着目画素Ｐ00で生じた階調誤差E00がレジスタR1ないしレジスタR4 の４つのレジスタに蓄積される様子を説明する。４つのレジスタの中、レジスタR1 は、着目画素から右横の画素に拡散される誤差が上書きされるレジスタである。他の３つのレジスタとは異なり、常に着目画素の右横の画素への誤差が上書きされる。図１２（a）ではレジスタR1 には、着目画素Ｐ00から右横の画素Ｐ01に拡散される誤差E00__R の値が上書きされている。他の３つのレジスタには、着目画素から左下、真下、右下の各画素に拡散されるそれぞれの誤差が加算される。いずれのレジスタにいずれの誤差が加算されるかは、以下に説明するように着目画素の移動に伴って順番に切り替わっていく。

【０１０５】先ず、着目画素が画素Ｐ00にある場合（図１２（a）参照）には、レジスタR2 には着目画素から画素Ｐ10への誤差（着目画素から真下の画素に拡散される誤差）が加算される。レジスタR3 には着目画素から画素Ｐ1-1 への誤差（着目画素から左下の画素に拡散される誤差）が加算される。レジスタR4 には着目画素から画素Ｐ11への誤差（着目画素から右下の画素に拡散される誤差）が加算される。

【０１０６】次に、着目画素が画素Ｐ01に移動した場合を考える（図１２（b）参照）。着目画素が移動しても、レジスタR2 には依然として画素Ｐ10への誤差が加算される。着目画素の移動後は、画素Ｐ10は着目画素の左下の画素となっているから、着目画素との位置関係で言えば、レジスタR2 には着目画素から左下の画素への誤差が加算されることになる。換言すれば、着目画素が画素Ｐ00にある時には、レジスタR2 には着目画素から真下の画素への誤差が加算されたのに対して、着目画素が画素Ｐ01に移動すると、着目画素から左下の画素への誤差が加算されることになる。

【０１０７】同様に、レジスタR4 には画素Ｐ11への誤差が加算されるので、着目画素が画素Ｐ00にあるときには着目画素から右下の画素への誤差が加算されるが、着目画素が画素Ｐ01に移動すると、着目画素から真下の画素への誤差が加算されることになる。

【０１０８】レジスタR3 については、着目画素が画素Ｐ00にあるときには着目画素の左下の画素Ｐ1-1 への誤差を加算したが、着目画素が画素Ｐ01に移動した後は、もはや画素Ｐ1-1 への誤差の拡散は不要である。そこで、レジスタR3 には新たな着目画素Ｐ02から画素Ｐ12への誤差を加算する。結局レジスタR3 についても他のレジスタと同様に、着目画素が移動すると、着目画素から見て異なる方向への誤差が加算されることになる。すなわち、着目画素が画素Ｐ00にある時には、レジスタR3 には着目画素から左下の画素への誤差が加算されるが、着目画素が画素Ｐ01に移動すると、着目画素から右下の画素への誤差が加算されることになる。

【０１０９】更に進んで、着目画素が画素Ｐ01から画素Ｐ02に移動した場合にも同様に、レジスタR2 ないしレジスタR4 の３つのレジスタには、着目画素の移動に伴って、着目画素から見て異なる方向にある画素への誤差が加算されることになる。図１２（c）は、着目画素が画素Ｐ02に移動した時に、着目画素Ｐ02から各レジスタに誤差が拡散されている様子を示している。前述の図１２（b）と図１２（c）とを比較すれば明らかなように、着目画素が画素Ｐ01にある時には、レジスタR2 ，レジスタR3 ，レジスタR4 の各レジスタには着目画素から左下の画素，右下の画素，真下の画素への誤差が加算されるが、着目画素が画素Ｐ02から画素Ｐ03に移動すると、各レジスタにはそれぞれ着目画素から右下の画素，真下の画素，左下の画素への誤差が加算されることになる。

【０１１０】このように、第２実施例の階調数変換処理においては、着目画素から右下の画素，真下の画素，左下の画素への誤差が、レジスタR2 ないしレジスタR4 の３つのレジスタに加算されるが、それぞれの誤差がいずれのレジスタに加算されるかは、着目画素の移動に伴って順番に切り替わっていく。こうして着目画素を移動させながら、各レジスタをそれぞれ適切なタイミングでリセットしつつ、順番に誤差を加算していけば、以下に説明するように、連続する３つの着目画素からの誤差が、各レジスタに順番に蓄積されることになる。以下、レジスタR4 を例にとって具体的に説明する。

【０１１１】着目画素が画素Ｐ00にある場合には、図１２（a）に示すようにレジスタR4には着目画素Ｐ00から右下の画素への誤差E00　RDが加算される。後述するように、誤差E00__RDを加算する時点では、レジスタR4 は既にリセットされているので、記憶されている誤差はない。着目画素が画素Ｐ01に移動すると、図１２（b）に示すように、レジスタR4 には着目画素Ｐ01から真下の画素への誤差E01　D が加算される。レジスタR4 には、先の着目画素Ｐ00からの誤差E00__RDが既に加算されているから、この誤差に加えて、着目画素Ｐ01からの誤差E01　Dが加算されることになる。着目画素が画素Ｐ02に移動すると、図１２（c）に示すように、レジスタR4 には着目画素Ｐ02から左下の画素への誤差E02__LDが加算される。レジスタR4 には、先の着目画素Ｐ00およびＰ01からの、それぞれの誤差E00　RDおよび誤差E01__D が既に加算されているから、これらの誤差に加えて着目画素Ｐ02からの誤差E02__LDが加算されることになる。その結果、着目画素Ｐ02からの誤差を加算した時点でレジスタR4 には、着目画素Ｐ00ないし着目画素Ｐ02の連続する３つの着目画素からの誤差が蓄積されることになる。図１２（c）において、レジスタR

４の右横に示された星印は、連続する３つの着目画素の中の最後の着目画素からの誤差が、レジスタR4 に蓄積されることを示している。こうして、連続する３つの画素からの誤差をレジスタR4 に蓄積し終わったら、蓄積した値を誤差バッファに書き込む。

【０１１２】上述の説明から明らかなように、レジスタR4 には画素P11に拡散される誤差が蓄積されているから、レジスタR4 に蓄積された値は、画素P11に対応する誤差バッファに書き込まれる。図１２（c）において、レジスタR4 から誤差バッファに向かう白抜きの矢印は、レジスタR4 に蓄積した誤差を誤差バッファに書き込む様子を模式的に示したものである。また、誤差バッファと画素P11との結ぶ破線の矢印は、画素P11に対応する誤差バッファであることを模式的に示したものである。蓄積した値を誤差バッファに書き込んだ後は、レジスタR4 の値をリセットする。図１２（a）の説明において、着目画素P00からの誤差を加算する時点では、レジスタR4 が既にリセットされていたのは、直前の着目画素P0-1 からの誤差を加算した時点でレジスタR4 に連続する３つの画素からの誤差が蓄積され、レジスタR4 がリセットされたからである。

【０１１３】次に、レジスタR3 に着目して説明する。図１２（a）に示すように、着目画素P00からの誤差を加算した時点で、レジスタR3 には連続する３つの着目画素P0-2 ，P0-1 ，P00からの誤差が蓄積されている。レジスタR3 の右横に示された星印は、着目画素P00からの誤差を蓄積すれば、連続する３つの着目画素からの誤差が蓄積されることを示している。そこで、着目画素P00からの誤差を加算した後に、レジスタR3 に蓄積された値を誤差バッファに書き込む。レジスタR3 に蓄積された誤差は画素P1-1 への誤差であるので、蓄積された誤差は、画素P1-1 に対応する誤差バッファに書き込む。こうして、蓄積した誤差を誤差バッファに書き込んだら、レジスタR3 の値をリセットしておく。

【０１１４】ここで、レジスタR3 がリセットされるタイミングと、前述のレジスタR4 がリセットされるタイミングとを比較すると、レジスタR4 は着目画素P00からの誤差が加算される前、すなわち着目画素P0-1 からの誤差を加算した後にリセットされるが、レジスタR3 は着目画素P00からの誤差を加算した後にリセットされる。従って、レジスタR3 はレジスタR4 に対して、着目画素１つ分だけ遅れてリセットされることになる。

【０１１５】図１２（b）に示すように、着目画素が画素P00から画素P01に移動すると、レジスタR3 には、それまでの画素P1-1 に代えて新たな画素P12が割り振られる。画素P12は着目画素P01に対して右下に位置する画素であるから、レジスタR3 には、着目画素から右下の画素へ拡散される誤差E01_RDが加算されることになる。レジスタR3 の値は先にリセットされているので、着目画素P01からの誤差が加算される時点では、レジスタR3 には加算されている誤差はない。着目画素が画素P02に移動すると、図１２（c）に示すように、レジスタR3 には、着目画素から真下の画素に拡散される誤差E02_D が、既に加算されている誤差E01_RDに加えて加算される。更に、着目画素が次の画素P03に移動した時点で、レジスタR3 には、画素P01ないし画素P03の連続する３つの着目画素からの誤差が蓄積されることになり、レジスタR3 に蓄積された値が誤差バッファに書き込まれることになる。前述したレジスタR4 に蓄積された誤差は、着目画素P02からの誤差を加算した後に誤差バッファに書き込まれたが、レジスタR3 については、画素１つ分だけ遅れて、着目画素P03からの誤差を加算した後に誤差バッファに書き込まれることになる。

【０１１６】レジスタR2 については、レジスタR3 に対して、更に着目画素１つ分だけタイミングが遅れることになる。すなわち、レジスタR3 については、図１２（a）に示すように、着目画素P00からの誤差を加算した後に、レジスタに蓄積した値が誤差バッファに書き込まれてレジスタがリセットされたが、レジスタR2 については、着目画素P01からの誤差を加算した後にレジスタに蓄積された値が誤差バッファに書き込まれてレジスタがリセットされることになる。

【０１１７】このように、レジスタR2 ないしレジスタR4 の３つのレジスタには、着目画素が移動する度に、連続する３つの着目画素からの誤差が順番に蓄積されて、誤差バッファに順次書き込まれていく。

【０１１８】以上、説明したように、４つのレジスタのうち、レジスタR1 は次の画素のドット形成有無の判断に使用する誤差を記憶し、レジスタR2 ないしレジスタR4の３つのレジスタは階調誤差を順次蓄積していくために使用すれば、１画素のドット形成有無を判断する度に、連続する３画素分の誤差の蓄積が終了した画素から、１画素分ずつ誤差バッファに書き込んでいくことができる。

【０１１９】Ｃ－２．第２実施例の階調数変換処理：図１３は、以上に説明した第２実施例の階調数変換処理の流れを示すフローチャートである。この処理も第１実施例の階調数変換処理と同様に、コンピュータ１００のＣＰＵ１０２によって行われる。尚、以下の説明では、インクの色あるいはドットの大きさを特定せずに説明するが、各色毎に、あるいは各種大きさのドットについて同様の処理を行う。以下では、図１３のフローチャートに従って、第１実施例の階調数変換処理との相違点を中心に、第２実施例の階調数変換処理について説明する。

【０１２０】第２実施例の階調数変換処理を開始すると、着目画素についての画像データCd と拡散誤差Ed とをＲＡＭ１０６から読み込み（ステップＳ４００）。続いて、画像データCd と、拡散誤差Ed と、レジスタR1 に記憶されている誤差とを加算することにより、補正データCx を算出する（ステップＳ４０２）。補正データCx を算出したら、レジスタR1 はリセットしておく（ステップＳ４０４）。こうして得られた補正データCx と所定の閾値ｔｈ とを比較して（ステップＳ４０６）、補正データの方が大きければドットを形成すると判断して、判断結果を示す変数Cr にドットを形成することを意味する値「１」書き込む（ステップＳ４０８）。そうでなければドットを形成しないと判断して、変数Cr にドットの形成しないことを意味する値「０」を書き込む（ステップＳ４１０）。

【０１２１】こうしてドット形成有無を判断したら、これに伴って発生する階調誤差Ｅを算出する（ステップＳ４１２）。階調誤差Ｅは、第１実施例と同様に、補正データCx から、着目画素での結果値（ドットを形成することにより、あるいはドットを形成しないことにより着目画素で表現される階調値）を減算することで求められる。

【０１２２】次いで、誤差拡散マトリックスによって画素位置毎に定まる所定の誤差拡散係数とステップＳ４１２で求めた階調誤差Ｅとを乗算し、画素位置毎に求められた誤差を、各レジスタに加算していく。先ず、着目画素から右隣の画素に拡散される誤差をレジスタR1 に加算する（ステップＳ４１４）。次いで、レジスタR2ないしレジスタR4 の３つのレジスタにそれぞれの誤差を加算していく。これら３つのレジスタの中で、連続する着目画素の３つ目の画素からの誤差が加算されるレジスタ、すなわち図１２の中で星印を付して示したレジスタには、予めフラグを設定しておく。こうしてフラグの設定されているレジスタには、着目画素の左下の画素に拡散する誤差を加算する（ステップＳ４１６）。図６（a）の誤差拡散マトリックスに示されるように、着目画素の左下の画素には、階調誤差Ｅと誤差拡散係数K1-1 とを乗算した値が配分される。誤差を加算していく３つのレジスタのうち、フラグの設定されたレジスタの１つ前のレジスタには、着目画素の直ぐ下の画素に配分すべき誤差を加算する（ステップＳ４１８）。着目画素の真下の画素に配分する誤差は、階調誤差Ｅと誤差拡散係数K10とを乗算して求められる。ここで、フラグの設定されたレジスタの１つ前のレジスタとは、例えば、図１２(a)に示すように星印がレジスタR3 に設定されている場合はレジスタR2 、図１２（b）に示すように星印がレジスタR2 に設定されている場合はレジスタR4 のレジスタを指す。３つのレジスタの中の残りのレジスタには、着目画素の右下の画素に配分すべき誤差、すなわち階調誤差Ｅと誤差拡散係数K11とを乗算した値を加算する（ステップＳ４２０）。

【０１２３】こうして階調誤差Ｅに所定の誤差拡散係数を乗算した値を、各レジスタに加算したら、フラグのセットされているレジスタに蓄積されている拡散誤差を、着目画素の左下の画素に対応する誤差バッファに書き込む（ステップＳ４２２）。例えば、図１２（c）に示すように着目画素が画素P02にある場合は、画素P11に対応する位置の誤差バッファに誤差を書き込んでやる。フラグのセットされたレジスタR4 には画素P00と画素P01と画素P02のそれぞれの画素から配分されてきた誤差が蓄積されており、図５に示した通常の誤差拡散法と比較すれば、第２実施例の処理においても実質的に同等の処理が行われていることが分かる。こうして、蓄積した誤差を誤差バッファに書き込んだら、新たに配分されてくる誤差を蓄積するためにレジスタをリセットしておく（ステップＳ４２４）。

【０１２４】以上のようにして１つの着目画素についての処理を終了したら、フラグを現在のレジスタの１つ前のレジスタに移動させた後（ステップＳ４２６）、全画素について処理を終了したか否かを判断し（ステップＳ４２８）、未処理の画素が残っていればステップＳ４００に戻って、全画素についての処理が終了するまで続く一連の処理を繰り返す。全画素についての処理が終了したら、第２実施例の階調数変換処理を抜けて、図４の画像データ変換処理に復帰する。

【０１２５】尚、ステップＳ４１４において、誤差をレジスタR1 に加算する代わりに、レジスタR1 に上書きすることとしてもよい。こうすれば、ステップＳ４０４においてレジスタR1 をリセットする処理を省略することができるので好ましい。同様に、ステップＳ４２０において、残りのレジスタに誤差を加算する処理を上書きする処理に変更すれば、ステップＳ４２４においてレジスタをリセットする処理を省略することが可能となるので好ましい。

【０１２６】以上、説明した第２実施例の階調数変換処理においては、１つの画素のドット形成有無を判断する度に、１つの画素の拡散誤差を誤差バッファに書き込むだけでよい。従って、通常の誤差拡散法に対して、誤差バッファに誤差を拡散するための時間を大きく短縮化することができる。前述したように、階調数変換処理に通常の誤差拡散法を用いた場合、誤差の拡散に要する時間は階調数変換処理に要する時間の中で比較的大きな割合を占めている。このことから、第２実施例の階調数変換処理を採用すれば処理時間が短縮化され、延いては画像を迅速に印刷することが可能となる。

【０１２７】尚、前述の第１実施例の階調数変換方法を用いた場合でも、誤差を拡散するために要する時間を短縮化することができるが、第２実施例の方法を採用すれば、誤差の拡散時間を更に効率よく短縮化することができる。すなわち、通常の誤差拡散法と比較した場合、前述の第１実施例の方法を使用した場合、１回のドット形成判断あたりでは２．５画素分の頻度で誤差を誤差バッファに加算するだけで足りていたが、第２実施例の方法を使用すれば、１回のドット形成判断あたりに誤差を書き込む頻度を１画素分にまで減少させることができる。

また、図7に示した第1実施例の場合では、中間バッファとして6つのレジスタを使用したが、第2実施例の方法の場合は4つのレジスタを使用しているだけであり、その分だけCPU102のレジスタを他の目的に使用することが可能になる。

【0128】C－3．変形例：上述した第2実施例の階調数変換処理においては、着目画素で発生した誤差を4つのレジスタに蓄積しているが、このうちのレジスタR1と、他の3つのレジスタR2ないしレジスタR4とでは、レジスタの使い方が若干異なっている。すなわち、レジスタR1については、着目画素の右隣の画素に配分すべき誤差が常に上書きされるが、他の3つのレジスタについては、着目画素に対する相対位置は一定ではなく、着目画素が移動する度に移動していた。

【0129】これに対して、図14に示すように、各レジスタの対応する画素位置を着目画素に対して固定することとしてもよい。すなわち、例えば、レジスタR1には着目画素の右隣の画素に配分すべき誤差が常に上書きされ、レジスタR2には着目画素の右下の画素に配分すべき誤差が、レジスタR3には着目画素の真下の画素に配分すべき誤差が、レジスタR4には着目画素の左下の画素に配分すべき画素が常に加算されるようにしてもよい。このように、各レジスタと着目画素との位置関係を固定した上で、後述するように、着目画素の移動に合わせて各レジスタに蓄積されている誤差を、順次となりのレジスタに移し替えていくのである。こうすれば、常に同じレジスタに蓄積されている誤差を誤差バッファに書き込めがよいので、レジスタから誤差バッファに拡散誤差を書き込む処理を簡素化することができる。また、各レジスタで使用される誤差拡散係数も常に同じ値となるので、着目画素で発生した階調誤差から各レジスタに加算する誤差を算出する処理も、簡素化することが可能となる。

【0130】以下、このような第2実施例の変形例としての階調数変換処理について、図15を参照しながら簡単に説明する。図15（a）は、着目画素P00についてのドット形成有無を判断した状態を示している。図の右側に示されている4つの矩形は、各レジスタを模式的に示したものである。レジスタR1には、常に着目画素の右側の画素に配分すべき誤差が上書きされる。また、レジスタR2には、常に着目画素の右下の画素に配分すべき誤差が加算され、レジスタR3には着目画素の真下の画素に配分すべき誤差が、レジスタR4には着目画素の左下の画素に配分すべき誤差が加算される。

【0131】このように、各レジスタには、着目画素に対して常に同じ位置の画素に配分すべき誤差が加算されるので、それぞれの誤差拡散係数は常に同じ値となる。すなわち、レジスタR1の誤差拡散係数は、常に着目画素の右側の画素への誤差拡散係数K01であり、レジスタR2の誤差拡散係数は、常に着目画素の右下の画素への誤差拡散係数K11である（図6参照）。同様に、レジスタR3の誤差拡散係数は常にK10であり、レジスタR4の誤差拡散係数は常にK1-1となる。こうして、各レジスタの誤差拡散係数が常に同じ値となるので、着目画素で発生した階調誤差から、各レジスタに加算すべき誤差の算出を容易に行うことができる。

【0132】着目画素で発生した階調誤差E00を各レジスタに加算したら、レジスタR4に蓄積されている誤差の値を誤差バッファに書き込む処理を行うと同時に、各レジスタに蓄積されている値を1つずつとなりのレジスタにシフトさせる。すなわち、レジスタR3の値をレジスタR4に移動させ、レジスタR2の値をレジスタR3に移動させる（図15（a）の下方を参照）。

【0133】以上の処理を終了したら、次いで、着目画素を画素P01に移して、再び同様の処理を行う。着目画素P01で発生した階調誤差E01を、所定の誤差拡散係数をかけてそれぞれのレジスタに加算したら、レジスタR4に蓄積されている値を誤差バッファに書き込む処理を行う。先に、各レジスタに記憶されている値を1つずつシフトする処理を行っているので、図15（b）に示すように各レジスタへ誤差を加算する処理が終了した時点では、レジスタR4には常に3つの画素からの誤差が蓄積されていることになる。従って、第2実施例の変形例においては、常に同じレジスタに蓄積されている誤差を誤差バッファに書き込んでやればよい。このため、レジスタから誤差バッファに拡散誤差を書き込む処理を簡素化することができる。尚、図15に示した処理では、各レジスタに蓄積されている誤差をとなりのレジスタにシフトさせる処理が新たに加わるが、このようなレジスタ間で値を移動させる処理はたいへん迅速に行うことができるので、そのことによる処理時間の増加は小さなものである。

【0134】上述した第2実施例の階調数変換処理あるいは第2実施例の変形例の階調数変換処理においては、説明が煩雑になるのを避けるために、いずれも誤差拡散マトリックスは誤差の拡散範囲の最も狭い図6（a）のマトリックスを使用するものとして説明した。もっとも、使用する誤差拡散マトリックスは図6（a）のマトリックスに限定されることなく、他のマトリックスを使用しても良いことはもちろんである。また、第1実施例における場合と同様に、キャッシュメモリを活用することにより、実質的に同様の処理を行っても良いことはもちろんである。

【0135】また、第2実施例の階調数変換方法においても、周辺の画素に拡散すべき中間的な拡散誤差をレジスタに加算しておくのではなく、階調誤差そのものを中間バッファに記憶しておき、新たな階調誤差を記憶する度に、誤差バッファに書き込むべき最終的な拡散誤差を算出するようにしても良い。かかる方法は、レジスタに記憶される値が蓄積中の拡散誤差であるか、階調誤差であるかの違いがあるものの、誤差バッファに書き込まれる誤差は同じ値となるので、実質的に同等の処理を行うことができる。

【0136】D．第3実施例：以上に説明した各実施例では、常に同じ誤差拡散マトリックスを使用して誤差を拡散するものとして説明したが、現実の階調数変換処理においては、画質上の要請から、図6（a）のような拡散範囲の狭い誤差拡散マトリックスと、図6（c）のような拡散範囲の広いマトリックスとを切り替えて使用する場合がある。

【0137】すなわち、誤差拡散法では、ドットが特定の周期的なパターンで形成されることを防ぐために、複数種類の誤差拡散マトリックスをランダムに切り替えながら使用することがある。また、特開平7－226841号には、画像データの階調値が十分に小さな所定の閾値よりも小さく、かつドットを形成すると判断されている場合に、拡散範囲の広い誤差拡散マトリックスを用いて誤差を拡散させることによって、ドット密度が疎な領域での、ドットの分散性を改善する技術が開示されている。こうすることで、ドットの分散性を改善可能な理由については説明を省略するが、ここでは、特開平7－226841号に開示された技術に従って、画像データの階調値が十分に小さな所定の閾値ｔｈｍよりも小さく、かつドットを形成すると判断されている場合には、図6（c）に示した誤差拡散範囲の広いマトリックスを使用し、それ以外の場合には図6（a）に示した誤差拡散範囲の狭いマトリックスを使用するものとする。このような場合に、以下に説明する第3実施例の方法を使用すれば、階調数変換処理に要する時間を効果的に短縮化することが可能となる。

【0138】図16は、2つの誤差拡散マトリックスを切り替えながら階調数変換処理を行う処理の流れを示したフローチャートである。以下、図16のフローチャートに従って、第3実施例の階調数変換処理について説明する。尚、上述した各実施例の階調数変換処理と同様、説明が煩雑化することを避けるために、ドットの色やドットの大きさを区別せずに説明するが、以下の処理は各色インク毎に、あるいは各種大きさのドット毎に行われる。

【0139】第3実施例の階調数変換処理を開始すると、先ず初めに着目画素の画像データCdと拡散誤差Edとを読み出す（ステップS500）。画像データCdと拡散誤差EdとはRAM106に記憶されている。レジスタR1に記憶されている左隣の画素からの誤差と、画像データCdおよび拡散誤差Edとを加算して補正データCxを算出する（ステップS502）。求めた補正データCxと所定の閾値ｔｈとを比較して（ステップS504）、補正データCxの方が大きければ着目画素にドットを形成すると判断し（ステップS504：ｙｅｓ）、ドット形成判断の結果を表す変数Crにドットを形成することを意味する値「1」を書き込んだ後（ステップS506）、ドットを形成したことにより着目画素に生じる階調誤差を算出する（ステップS508）。補正データCxが所定の閾値ｔｈよりも小さければ（ステップS504：ｎｏ）、変数Crにドットを形成しないことを意味する値「0」を書き込んで（ステップS510）、そのことにより着目画素に生じる階調誤差を算出する（ステップS512）。

【0140】着目画素にドットを形成している場合には、着目画素の画像データCdと誤差拡散マトリックスを切り替えるための所定の閾値ｔｈｍとを比較する（ステップS514）。画像データCdが閾値ｔｈｍよりも小さい場合は（ステップS514：ｎｏ）、小さな画像データの領域にたまたまドットが形成されたものと考えられるので、誤差拡散範囲の広い誤差拡散マトリックス（ここでは、図6（c）に示したマトリックス）に従って、それぞれの誤差拡散係数を設定する（ステップS516）。すなわち図6（c）に示すように、着目画素の真下の画素の誤差拡散係数K10には1／4を設定し、着目画素の右隣の画素、右下の画素、および左下の画素のそれぞれの誤差拡散係数K01、K11、K1-1には1／8を設定する。着目画素から更に遠くにある画素の誤差拡散係数K02、K03、K12、K13、K1-2，K1-3には1／16を設定する。

【0141】次いで、着目画素から遠方にある6つの画素については、階調誤差と各画素の誤差拡散係数とを乗算した値をそれぞれの誤差バッファに直接加算する（ステップS518）。これを、図17を用いて説明する。図17は着目画素P00についてドット形成有無を判断し、その結果、階調誤差E00が発生した状態を示している。この階調誤差E00は、図6（c）に示す誤差拡散マトリックスを用いて広い範囲に拡散する。すなわち、通常は図17中に太い破線で囲った4つの画素に誤差を拡散させるが、画素P00の階調誤差E00は破線の外側にある画素P02、画素P03、画素P12、画素P13、画素P1-2、および画素P1-3の6つの画素にも誤差を拡散させる。ステップS518の処理においては、これら6つの画素について、階調誤差E00とそれぞれの画素の誤差拡散係数とを乗算した値を、各画素の誤差バッファに直接加算するのである。

【0142】遠方の画素の誤差バッファに誤差を加算したら、破線の内側にある各画素のレジスタに誤差を拡散する（ステップS522）。すなわち、第3実施例の階調数変換処理においては、着目画素の遠方の画素については誤差バッファに直接誤差を加算するが、着目画素周辺の各画素については前述の第1実施例あるいは第2実施例と同様に中間バッファを利用して誤差を拡散するのである。以下では、第2実施例の変形例の方法に準じて誤差を拡散する場合、すなわち各レジスタには、着目画素に対して常に同じ位置関係にある画素への誤差が加算

される場合を例にとって説明する。

【０１４３】先ず、着目画素の右側の画素への誤差拡散係数K01と階調誤差E00とを乗算した値で、レジスタR1の値を更新する。次に、誤差拡散係数K11と階調誤差E00とを乗算した値をレジスタR2に加算し、誤差拡散係数K01と階調誤差E00とを乗算した値をレジスタR3に加算し、誤差拡散係数K1-1と階調誤差E00とを乗算した値をレジスタR4に加算する（図１５（a）を参照）。ステップＳ５２２では以上の処理を行う。続いて、レジスタR4に蓄積されている値を対応する画素の誤差バッファに加算する処理を行う（ステップＳ５２４）。図１７に示すように、ここでは着目画素は画素P00であるから、画素P1-1の誤差バッファにレジスタR4の値を加算すればよい。誤差バッファへ加算する処理が終了したら、レジスタR3に蓄積されている値をレジスタR4に、レジスタR2に蓄積されている値をレジスタR3にシフトさせる（ステップＳ５２６）。各レジスタの値をシフトする処理を行った後、レジスタR2の値をリセットしておく（ステップＳ５２８）。

【０１４４】一方、着目画素にドットを形成していない場合（ステップＳ５０４：ｎｏ）、あるいは着目画素にドットを形成していても画像データCdが所定の閾値ｔｈｍより大きい場合（ステップＳ５１４：ｙｅｓ）には、誤差拡散範囲の狭い方のマトリックス（ここでは図６（a）に示したマトリックス）に従って、それぞれの誤差拡散係数を設定する（ステップＳ５１８）。すなわち図６（a）の誤差拡散マトリックスに従って、着目画素周辺の４つの画素の誤差拡散係数に１／４を設定する。こうして設定された誤差拡散係数を用いて、上述したステップＳ５２２ないしステップＳ５２８の処理を行うことにより、レジスタR4に蓄積された誤差が誤差バッファに加算される。

【０１４５】以上のようにして着目画素のドット形成有無を判断し、１画素についての拡散誤差を誤差バッファに書き込んだら、すべての画素について処理を終了したか否かを判断する（ステップＳ５３０）。未処理の画素が残っている場合は、再びステップＳ５００に戻って、新たな着目画素についての画像データと拡散誤差とをＲＡＭ１０６から読み込み、これらの値とレジスタR1に記憶されている誤差とを用いて補正データCxを算出する（ステップＳ５０２）。以降、未処理の画素がなくなるまで以上のような一連の処理を繰り返し、全画素について処理を終了したら、第３実施例の階調数変換処理を抜けて図４に示す画像データ変換処理に復帰する。

【０１４６】上述した第３実施例の方法を用いて階調数変換処理を行えば、拡散範囲の広い誤差拡散マトリックスと、拡散範囲の狭い誤差拡散マトリックスとを切り替えながら階調数変換処理を行う場合に、処理時間を効率よく短縮化することができる。すなわち、図６（c）に示した拡散範囲の広い誤差拡散マトリックスを使用する場合、すべての誤差をレジスタに蓄積しようとすると、多数のレジスタが必要となる。図１７を参照して説明すると、破線で囲った範囲の内側にある４画素分のレジスタと、破線の外側の６画素分のレジスタとを加えた１０画素分のレジスタを使用する。これらレジスタに誤差を加算するために、フラグを使用する第２実施例の方法を用いるにせよ、レジスタの値をシフトさせる第２実施例の変形例の方法を用いるにせよ、レジスタの数があまりに多くなれば、レジスタの操作に時間が必要となって、全体としての処理時間を増加させる。

【０１４７】これに対して、上述した第３実施例の方法を用いれば、たとえ非常に広い範囲に誤差を拡散するマトリックスを用いた場合でも、誤差を加算するレジスタの数を増加させずに階調数変換処理を行うことができるので、全体として処理が簡素化され、処理時間が増加することを回避することが可能となる。

【０１４８】特に、前述したように、拡散範囲の広い誤差拡散マトリックスは、画像データの階調値が十分小さく、かつドットを形成すると判断された場合に使用される。画像データの階調値が十分に小さければ、ドットを形成すると判断される確率は小さいことから、拡散範囲の広い誤差拡散マトリックスは使用頻度の低いマトリックスであると言える。従って、ごく稀に広い範囲に誤差を拡散しなければならない場合には、多少時間がかかっても誤差バッファに直接誤差を加算することとし、それほど広い範囲に拡散させる必要のない通常の場合は、中間バッファを活用して迅速に誤差を拡散させることで、全体として階調数変換処理を迅速に行うことができる。また、広い範囲に拡散する場合には、遠方の画素の誤差バッファには直接誤差を加算することとすれば、それだけ中間バッファとして使用するレジスタの数を節約することができ、浮いたレジスタを他の処理に使用することで、更に処理を効率化することが可能となる。

【０１４９】尚、上述の第３実施例においては、誤差拡散マトリックスを切り換えることとし、拡散範囲の広い誤差拡散マトリックスが選択されている場合にのみ、遠方の画素の直接誤差を拡散させるものとしたが、必ずしも、マトリックスの切換を行う場合に限定されるものではない。例えば、使用する誤差拡散マトリックスが拡散範囲の広いマトリックスである場合に、遠方の画素には直接誤差を拡散して記憶させ、近傍の画素には、中間バッファを活用することにより、所定画素数の階調誤差に基づき、まとめて誤差を拡散させるようにしても良い。こうすれば、まとめて誤差を拡散して記憶させるために必要な中間バッファを節約することが可能となる。

【０１５０】また、以上に説明した第３実施例においても、中間バッファを活用することによって、所定画素数の階調誤差から、周辺画素へ拡散誤差をまとめて拡散して記憶させるために方法には、第１実施例ないし第２実施例として説明した各種の方法を好適に適用可能であることは言うまでもない。

【０１５１】Ｅ．第４実施例：上述した各種実施例の方法は、最終的には、階調誤差を周辺画素の誤差バッファに配分している。この意味からは、いわゆる誤差拡散法と呼ばれる方法に類似する手法と考えることができる。もちろん、いわゆる平均誤差最小法と呼ばれる方法のように、ドット形成有無の判断によって発生した階調誤差を着目画素に記憶しておき、未判断画素のドット形成有無の判断に際しては、周辺画素から階調誤差を読み出してドット形成判断を行う場合にも、中間バッファを活用することでドット形成判断に要する時間を短縮化することが可能である。以下では、このような方法を用いて階調数変換処理を行う第４実施例について説明する。

【０１５２】Ｅ－１．第４実施例において階調数変換処理の時間を短縮化する原理：図１８は、第４実施例の方法において、中間バッファを活用することによってドット形成有無の判断に要する時間を短縮化する原理を示す説明図である。図１８（a）は、着目画素P00についてドットの形成有無を判断している様子を示している。第４実施例の方法を説明する準備として、いわゆる平均誤差最小法と呼ばれる方法を、図１８（a）を流用して簡単に説明しておく。

【０１５３】平均誤差最小法では、ドット形成有無の判断によって生じた階調誤差を、その画素に対応する誤差バッファに記憶しておく。これを、図１８（a）に則して説明すると、左側の図中に例えば「E0-1」とあるのは画素P0-1で発生した階調誤差を示しており、画素P0-1の枠内にE0-1と表示することによって、階調誤差E0-1が画素P0-1の誤差バッファに記憶されていることを模式的に示している。また、図中に斜線が付されているのは、ドット形成判断済みであることを示している。図示されているように、平均誤差最小法では、ドット形成有無の判断済みの画素にそれぞれの階調誤差が記憶されている。未判断の着目画素P00についてドット形成有無を判断する場合には、周辺の判断済みの画素からそれぞれの階調誤差を読み出して、これらの誤差を考慮しながら着目画素P00のドット形成有無を判断する。より詳細には、図１９にいくつか例示されているように、周辺画素の着目画素に対する相対位置に応じて所定の重み係数が予め定められており、周辺の各画素から読み出した誤差に所定の重み係数をかけた値で、着目画素の画像データを補正することにより補正データを算出する。尚、図１９では斜線が付されている画素が着目画素であり、各画素に表示されている数値がその画素に設定されている重み係数である。こうして求めた補正データと所定の閾値とを比較することによって、着目画素についてのドット形成有無を判断する。着目画素についてドット形成有無を判断したら、そのことで生じる階調誤差を算出し、算出した階調誤差を着目画素の誤差バッファに記憶する。平均誤差最小法では、以上のような処理を繰り返すことによって、画素毎にドットの形成有無の判断を行う。

【０１５４】上述したように、平均誤差最小法を用いてドット形成有無を判断するためには、１つの画素についての判断を行う度に、誤差バッファから複数の画素の階調誤差を読み出さなければならず、ドット形成有無の判断するためにある程度の時間が必要となる。

【０１５５】これに対して、以下に説明する第４実施例の階調数変換処理においては、数学的には上述の平均誤差最小法と等価な処理を行いながらも、中間バッファを活用することによって、ドット形成の有無を迅速に判断することが可能となっている。以下、図１８を参照しつつ、前述の平均誤差最小法と対比することにより、第４実施例の階調数変換処理において処理時間を短縮化する原理について説明する。尚、説明の煩雑化を避けるために、以下では、図１９（a）に示した重み係数の設定に従って、周辺画素の階調誤差を考慮するものとする。

【０１５６】図１８（a）は、第４実施例の階調数変換処理において着目画素P00についてのドット形成有無を判断している様子を示す説明図である。着目画素P00についてのドット形成有無の判断には、前述した平均誤差最小法と同様に、画素P-1-1で発生した階調誤差E-1-1と、画素P-10で発生した階調誤差E-10と、画素P-11で発生した階調誤差E-11と、画素P0-1で発生した階調誤差E0-1とを使用する。図１８（a）の右側の４つの矩形は、中間バッファとして使用される４つのレジスタを模式的に示したものである。説明の便宜から、ここでは各レジスタにR1，R2，R3，R4とそれぞれ符合を付して区別することにする。

【０１５７】各レジスタには、着目画素に対して所定の位置関係にある画素の階調誤差が上書きされる。すなわち、レジスタR1には、常に着目画素の右上にある画素での階調誤差が記憶され、レジスタR2には、着目画素の真上にある画素での階調誤差が記憶され、レジスタR3には、着目画素の左上の画素の階調誤差が、レジスタR4には左隣の画素の階調誤差が記憶される。各レジスタにそれぞれの階調誤差を記憶する処理については後述する。

【０１５８】第４実施例の階調数変換処理では、これら４つのレジスタに記憶されている各画素での階調誤差と、予め画素毎に設定されている重み係数とを考慮して、着目画素での補正データを算出する。つまり、前述した平均誤差最小法では、各画素での階調誤差を誤差バッファから読み出すのに対して、ここでは各レジスタに予め記憶されているところが大きく異なっている。こうして求めた補正データと所定の閾値とを比較することにより、着目画素についてのドット形成有無を判断する。ドット形成有無を判断したら、続いて着目画素P00での階調誤差E00を算出する。階調誤差は、補正データと着目画素での結果値との差を取ることによって求めること

ができる。

【０１５９】こうして、着目画素についての階調誤差E00が求められたら、新たな着目画素についてのドット形成判断を行うために、各レジスタに対して図１８（ｂ）に示す操作を加える。先ず、レジスタR4 に記憶されている誤差を誤差バッファに書き込む。図１８（ａ）を用いて説明したように、レジスタR4 には、常に着目画素の左隣の画素で生じた階調誤差が記憶されているので、レジスタR4 の値は着目画素の左隣の画素の誤差バッファに上書きしてやる。次いで、先程求めた画素P00についての階調誤差E00をレジスタR4 に上書きし、更にレジスタR2 の値をレジスタR3 に、レジスタR1 の値をレジスタR2 にそれぞれ移動させる。すなわち、着目画素が画素P00から右隣の画素P01に移動することに対応して、各レジスタの値を移動させるのである。これらの処理は、ＣＰＵ１０２内部のレジスタ間でデータを移動させるだけでよいので、極めて迅速に行うことができる。最後に、新たな着目画素の右上の画素での階調誤差を誤差バッファから読み出して、レジスタR1 に記憶させる。

【０１６０】以上のような操作を加えることにより、各レジスタに記憶されている値は、図１８（ａ）の右側に示した状態から図１８（ｃ）の状態となる。図１８（ａ）と図１８（ｃ）とを比較すれば明らかなように、図１８（ｃ）の各レジスタに記憶されている値は、図１８（ａ）において着目画素を画素P00から画素P01に変更したときの各レジスタの値となっている。従って、以上のような処理を繰り返し行えば、次々に新たな画素のドット形成判断を行うことができる。このように、第４実施例の方法では、続けて使用する階調誤差はレジスタに蓄えておくことによって、誤差バッファから階調誤差を読み出す頻度を大きく軽減させることが可能となるのである。

【０１６１】Ｅ－２．第４実施例の階調数変換処理：以下、上述した第４実施例の階調数変換処理を、実際に行うための処理の流れについて簡単に説明する。図２０は、第４実施例の階調数変換処理の流れを示したフローチャートである。尚、上述した各種実施例の階調数変換処理と同様に、説明の煩雑化を避けるために、以下ではインクの種類やドットの大きさを特定せずに説明するが、各色毎に、あるいは各種大きさのドット毎に同様の処理が行われる。

【０１６２】第４実施例の階調数変換処理を開始すると、先ず初めに着目画素の画像データCd を読み出し（ステップＳ６００）、続いて着目画素の右上の画素の階調誤差を誤差バッファから読み出してレジスタR1 に記憶する（ステップＳ６０２）。この結果、図１８を用いて説明したように、各レジスタには着目画素周辺の各画素での階調誤差が記憶される。

【０１６３】次いで、各レジスタに記憶されている誤差と所定の重み係数とを、レジスタ毎に乗算して、着目画素の補正データCx を算出する（ステップＳ６０４）。図１９に示したように、着目画素周辺の画素には、画素毎に所定の重み係数が定められているので、これら所定の重み係数と記憶されている誤差とをレジスタ毎に乗算し、得られた各乗算値と先に読み出した画像データCd とを加算することにより、着目画素の補正データCx を算出するのである。こうして求めた補正データCx と所定の閾値ｔｈとを比較する（ステップＳ６０６）。補正データCx の方が大きければ着目画素にドットを形成すると判断して、判断結果を示す変数Crにドットを形成することを意味する値「１」を書き込み（ステップＳ６０８）、そうでなければドットを形成しないことを意味する値「０」を書き込む（ステップＳ６１０）。続いて、かかる判断結果によって着目画素に生じる階調誤差を算出する（ステップＳ６１２）。階調誤差は、前述した各種実施例と同様に、着目画素の補正データから結果値を減算することで求められる。

【０１６４】以上のようにして着目画素での階調誤差が求められたら、図１８（ｂ）を用いて説明したように、各レジスタに次のような一連の操作を加える。先ず、レジスタR4 に記憶されている階調誤差を、誤差バッファに書き込む（ステップＳ６１４）。次ぎに、ステップＳ６１２で求めた階調誤差を、レジスタR4 に書き込む（ステップＳ６１６）。続いて、レジスタR2 の値をレジスタR3 に移動し（ステップＳ６１８）、レジスタR1 の値をレジスタR2 に移動させる（ステップＳ６２０）。

【０１６５】以上のようなレジスタの操作が終了したら、全画素についてドット形成判断を終了したか否かを判断し（ステップＳ６２２）、未判断の画素が残っていたら、ステップＳ６００に戻って、すべての画素について処理が終了するまで、続く一連の処理を繰り返す。すべての画素のドット形成有無を判断したら、第４実施例の階調数変換処理を抜けて、図４に示す画像データ変換処理に復帰する。

【０１６６】以上に説明した第４実施例の階調数変換処理においては、１画素のドット形成有無を判断する度に、誤差バッファから１画素分の階調誤差を読み出すだけでよい。平均誤差最小法を適用する場合には、前述したように、１画素の判断を行う度に誤差バッファから４画素分の拡散誤差を読み出す必要があることと比較して、誤差バッファに対してデータを読み書きする頻度を減らすことができ、その分、階調数変換処理を迅速に行うことができる。もちろん、第４実施例の方法と平均誤差最小法とは、数学的には全く等価な処理を行っており、第４実施例の方法を用いれば、平均誤差最小法を用いた場合と同様に、高画質の画像を得ることができる。

【０１６７】尚、上述した第４実施例においては、重み係数の設定として、図１９（ａ）に例示する値が設定されているものとして説明した。実際の重み係数は、図１９に例示する設定に限定されることなく、画質の要請に応じて種々の設定とすることができることは言うまでもない。また、上述の第４実施例では、図１９（ａ）に例示した設定を用いていることに対応して、中間バッファとして４つのレジスタを使用したが、使用する重み係数の設定に応じて、より多くのレジスタが必要になる場合があることはもちろんである。

【０１６８】更に、上述した第４実施例においては、中間バッファとして、ＣＰＵ１０２に内蔵されたレジスタを用いるものとして説明したが、レジスタに限らず、キャッシュメモリなどの高速に読み書き可能な記憶素子を用いても良いことはもちろんである。

【０１６９】以上、各種の実施例について説明してきたが、本発明は上記すべての実施例に限られるものではなく、その要旨を逸脱しない範囲において種々の態様で実施することができる。

【０１７０】例えば、上述の各種実施例においては、各画素の補正データと所定の閾値との大小関係に基づいてドットの形成有無を判断するものとして説明した。もちろん、ドット形成の有無を判断する方法は、前述した方法に限らず、周知の各種方法を適用することが可能である。

【０１７１】また、上述の各種実施例においては、説明の煩雑化を避けるために、形成されるドットの種類は１種類として、各画素にはドットが形成されるか、形成され無いかの２つの状態しか取り得ないものとして説明した。もちろん、ドットの大きさ、あるいはインク濃度の異なる複数種類のドットを形成可能としてもよい。例えば、大・小の２種類のドットを形成可能として、次のようにしてドットの形成有無を判断しても良い。すなわち、２つの閾値ｔｈ１とｔｈ２（ただし、ｔｈ１＞ｔｈ２とする）を設定しておき、画素の補正データが閾値ｔｈ１より大きければ大ドットを形成すると判断し、閾値ｔｈ１より小さくかつ閾値ｔｈ２より大きければ小ドットを形成すると判断し、閾値ｔｈ２より小さい場合はドットを形成しないと判断する。各画素で発生する階調誤差は、その画素の補正データから結果値を減算することによって算出することができる。

【０１７２】尚、以上の各種実施例においては、誤差バッファは論理的には各画素毎に用意されることになるが、実際上は数ラスタ分の誤差バッファのみを用意して、ドット形成有無が判断された画素の誤差バッファは、他の画素の誤差バッファとして転用される。説明の煩雑化を避ける目的で、上述の各種実施例では、あたかも全画素分の誤差バッファが用意されているかのように説明したが、数ラスタ分の誤差バッファを繰り返し使用しても構わないのはもちろんである。

【０１７３】また、上述の機能を実現するソフトウェアプログラム（アプリケーションプログラム）を、通信回線を介してコンピュータシステムのメインメモリまたは外部記憶装置に供給し実行するものであってもよい。もちろん、ＣＤ－ＲＯＭやフレキシブルディスクに記憶されたソフトウェアプログラムを読み込んで実行するものであっても構わない。

【０１７４】また、上述した各種実施例では、階調数変換処理を含む画像データ変換処理はコンピュータ内で実行されるものとして説明したが、画像データ変換処理の一部あるいは全部をプリンタ側、あるいは専用の画像処理装置を用いて実行するものであっても構わない。

【０１７５】更には、画像表示装置は、必ずしも印刷媒体上にインクドットを形成して画像を印刷する印刷装置に限定されるものではなく、例えば、液晶表示画面上で輝点を適切な密度で分散させることにより、階調が連続的に変化する画像を表現する液晶表示装置であっても構わない。

【図面の簡単な説明】

【図１】本実施例の印刷システムの概略構成図である。

【図２】本実施例の画像処理装置としてのコンピュータの構成を示す説明図である。

【図３】本実施例の画像表示装置としてのプリンタの概略構成図である。

【図４】本実施例の画像処理装置で行われる画像データ変換処理の流れを示すフローチャートである。

【図５】誤差拡散法を用いてドットの形成有無を判断する様子を概念的に示す説明図である。

【図６】画素毎に誤差拡散係数が設定されている様子を例示する説明図である。

【図７】第１実施例の階調数変換処理において処理時間を短縮化する原理を示す説明図である。

【図８】第１実施例の階調数変換処理の流れを示すフローチャートである。

【図９】第１実施例の階調数変換処理において１度に多数の画素の誤差を拡散する場合を示す説明図である。

【図１０】第１実施例の変形例の階調数変換処理の流れを示すフローチャートである。

【図１１】第２実施例の階調数変換処理において処理時間を短縮化する原理を示す説明図である。

【図１２】第２実施例の階調数変換処理において中間バッファと誤差バッファとでデータを読み書きする処理の概要を示す説明図である。

【図１３】第２実施例の階調数変換処理の流れを示すフローチャートである。

【図１４】各レジスタの対応する画素位置が着目画素に対して一定とした場合に、各レジスタと着目画素周辺の画素とが対応している様子を例示する説明図である。

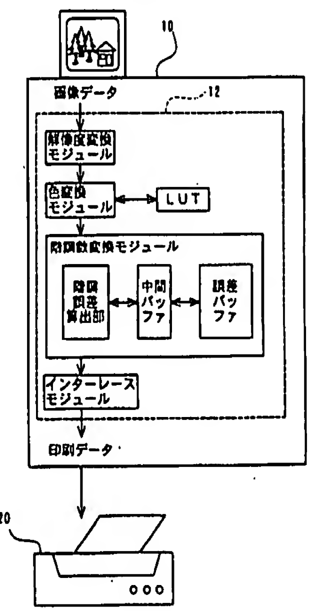
【図１５】第２実施例の変形例の階調数変換処理において処理時間を短縮化する原理を示す説明図である。

【図１６】第３実施例の階調数変換処理の流れを示すフローチャートである。

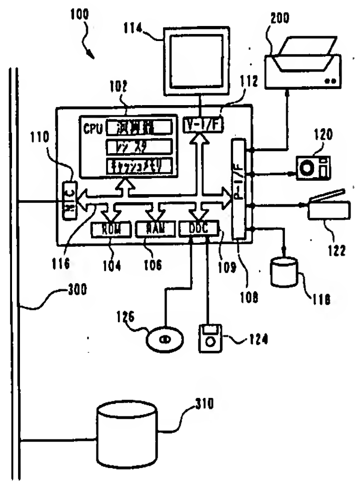
【図１７】第３の実施例の階調数変換処理において拡散範囲の広いマトリックスを使用した場合と、拡散範囲の狭いマトリックスを使用した場合とで、階調誤差が拡散される範囲を比較して示す説明図である。
【図１８】第４実施例の変形例の階調数変換処理において処理時間を短縮化する原理を示す説明図である。
【図１９】第４実施例の変形例の階調数変換処理において、重み係数が画素毎に設定されている様子を例示する説明図である。
【図２０】第４実施例の階調数変換処理の流れを示すフローチャートである。

【符号の説明】
１０…コンピュータ
１２…プリンタドライバ
２０…カラープリンタ
１００…コンピュータ
１０２…ＣＰＵ
１０４…ＲＯＭ
１０６…ＲＡＭ
１０８…周辺機器インターフェースＰ・Ｉ／Ｆ
１０９…ディスクコントローラＤＤＣ
１１０…ネットワークインターフェースカードＮＩＣ
１１２…ビデオインターフェースＶ・Ｉ／Ｆ
１１４…ＣＲＴ
１１６…バス
１１８…ハードディスク
１２０…デジタルカメラ
１２２…カラースキャナ
１２４…フレキシブルディスク
１２６…コンパクトディスク
２００…カラープリンタ
２３０…キャリッジモータ
２３５…紙送りモータ
２３６…プラテン
２４０…キャリッジ
２４１…印字ヘッド
２４２，２４３…インクカートリッジ
２４４…インク吐出用ヘッド
２６０…制御回路
２６１…ＣＰＵ
２６２…ＲＯＭ
２６３…ＲＡＭ
３００…通信回線
３１０…記憶装置

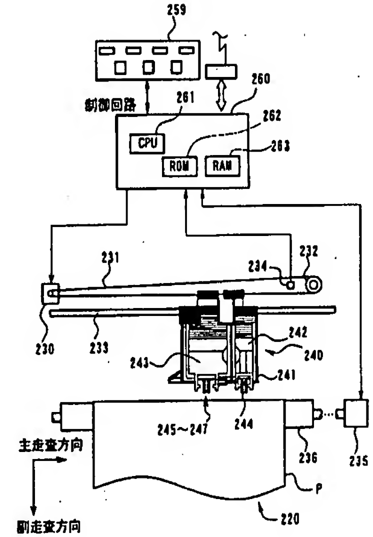
【図１】



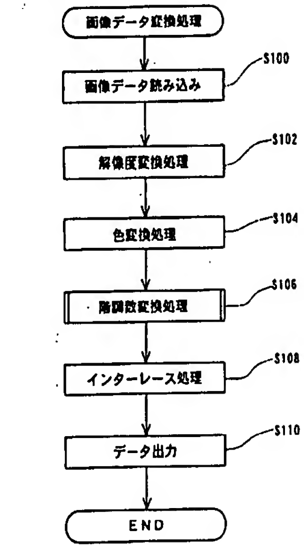
【図２】



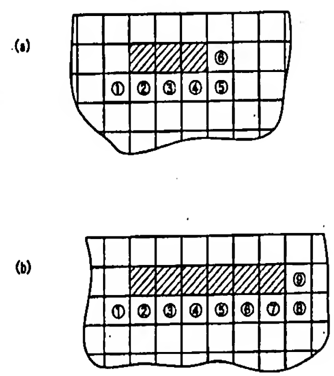
【図３】



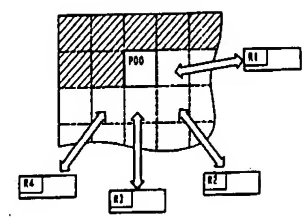
【図４】



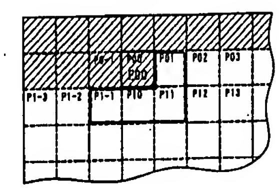
【図９】



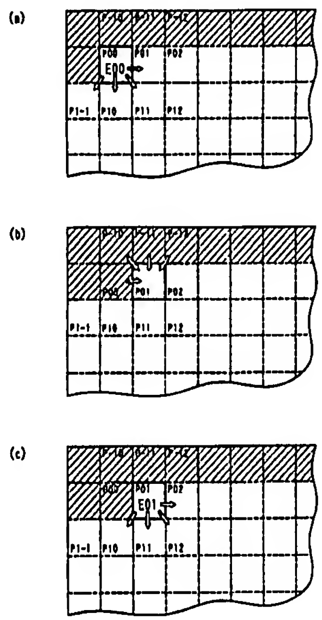
【図１４】



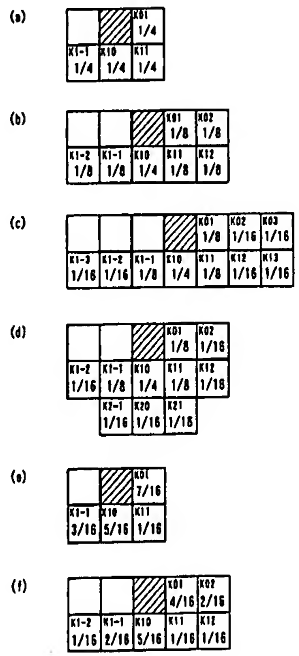
【図１７】



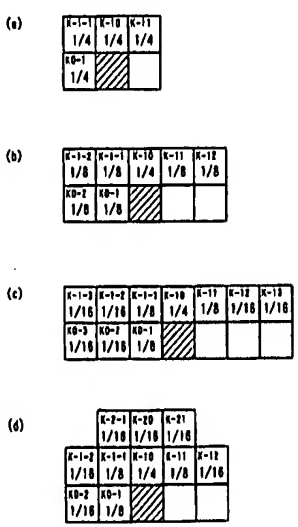
【図５】



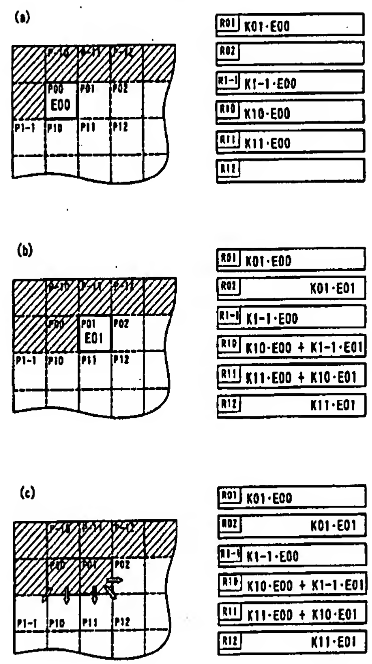
【図６】



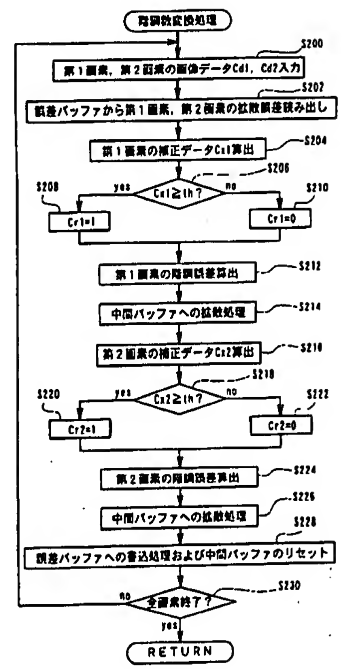
【図１９】



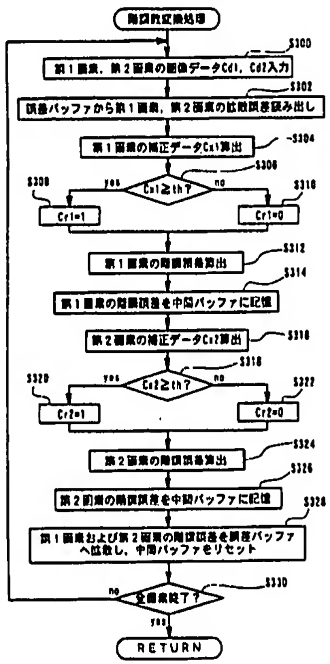
【図７】



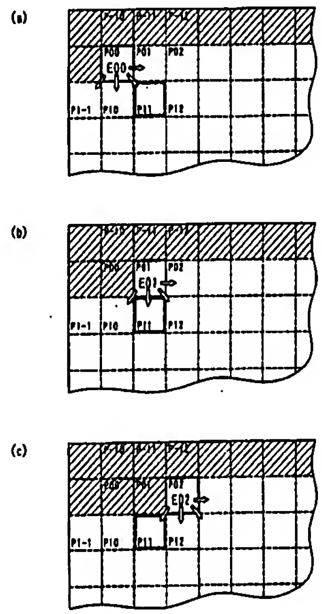
【図８】



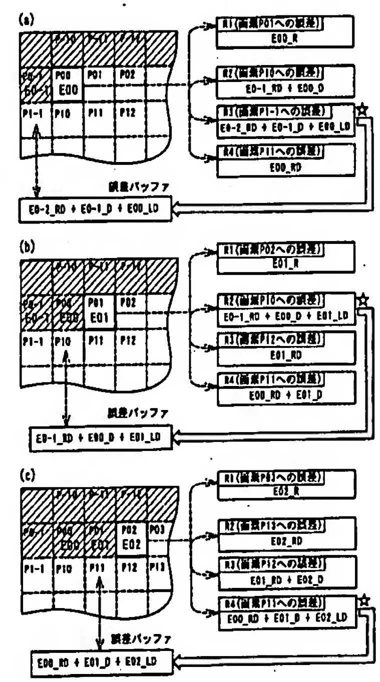
【図１０】



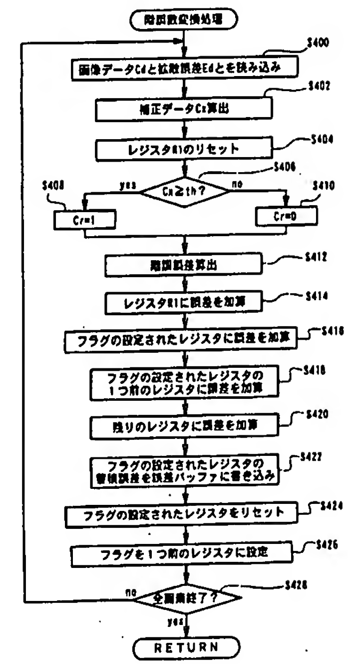
【図１１】



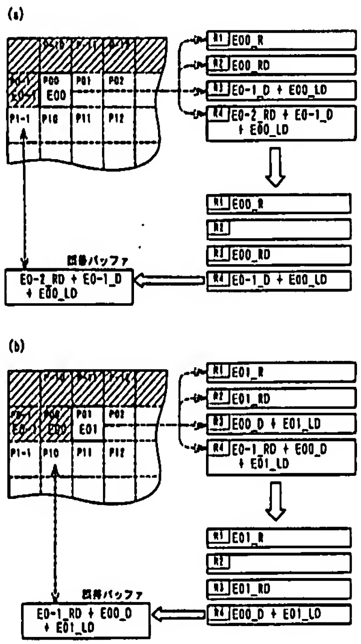
【図１２】



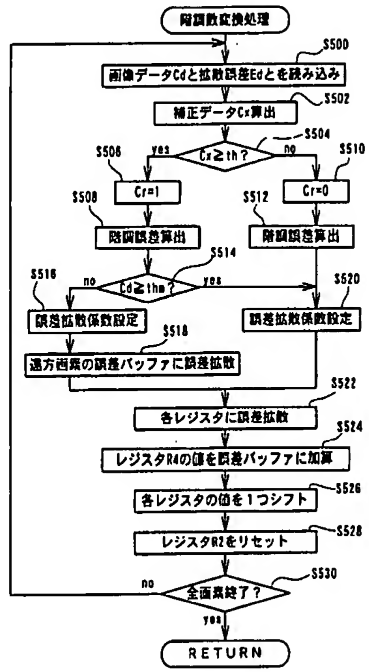
【図１３】



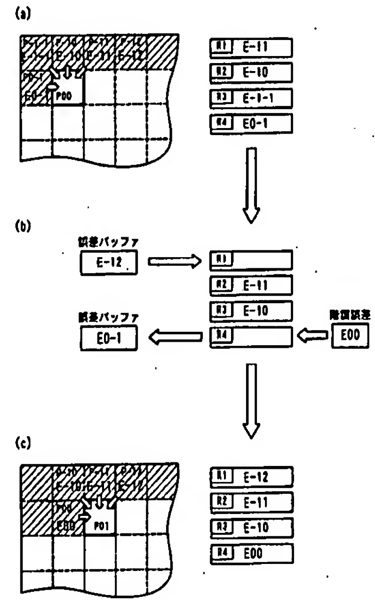
【図１５】



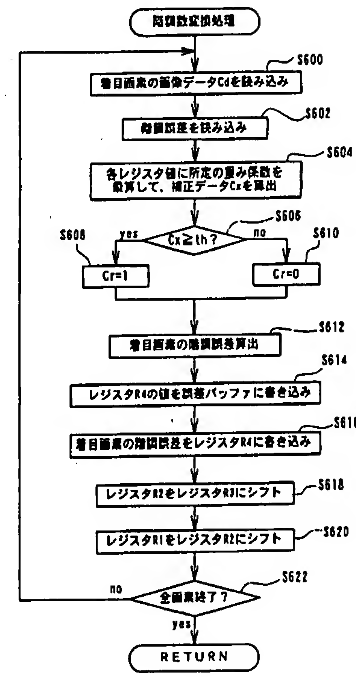
【図１６】



【図１８】



【図２０】



フロントページの続き

Fターム(参考) 2C262 AA24 AB19 AC07 BB01 BB08
BB22 DA09 EA04 EA06 GA11
GA12
5B021 AA01 LG08
5B057 AA11 CA01 CA08 CA12 CA16
CB01 CB07 CB12 CB16 CE13
CH01 CH11
5C077 LL18 MP01 MP08 NN11 PP47
PQ12 PQ22 TT02

CLAIMS

[Claim(s)]

[Claim 1] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel It is the image processing system which changes this image data into the image data of the transcription by the existence of dot formation. A gradation error maintenance means to compute the gradation error generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, and to hold it temporarily, It is based on said gradation error about two or more pixels of a predetermined number held temporarily. A diffusion error storage means to compute the diffusion error diffused in the non-judged pixel by which decision of dot formation existence is not made by being on the outskirts of two or more pixels of this predetermined number, to match the this computed diffusion error with this non-judged pixel, and to memorize it, The image processing system equipped with a dot formation decision means to judge the dot formation existence about this non-judged pixel, based on the gradation value of this non-judged pixel in said image data, taking into consideration said diffusion error memorized by matching with said non-judged pixel.

[Claim 2] Said gradation error maintenance means is an image processing system according to claim 1 which is the means which can be written quickly about said gradation error rather than it writes to said diffusion error storage means.

[Claim 3] It is an image processing system according to claim 1. Said diffusion error storage means A distribution error are recording means by which said gradation error held temporarily computes the distribution error distributed to each which is said non-judged pixel for this every gradation error, and accumulates it in this each sheep decision pixel, An image processing system equipped with a diffusion error are recording means to match said these all distribution errors accumulated with said non-judged pixel as said diffusion error whenever it accumulates said all distribution errors about two or more pixels of said predetermined number, and to accumulate them.

[Claim 4] It is the image processing system which is a means are an image processing system according to claim 1, and said gradation error maintenance means is a means hold said gradation error about two or more pixels of said predetermined number, matches with each of this sheep decision pixel all the diffusion errors that said diffusion error storage means computed said diffusion error diffused to each of said non-judged pixel based on the

gradation error of said predetermined number currently held, and were this computed, and accumulate.
[Claim 5] It is an image processing system according to claim 1. Said diffusion error storage means A distribution error are recording means by which said gradation error held temporarily computes the distribution error distributed to each which is said non-judged pixel for this every gradation error, and accumulates it in this each sheep decision pixel, An image processing system equipped with a distribution error storage means to match with this non-judged pixel the distribution error which is said each sheep decision pixel in which the distribution error for two or more pixels of said predetermined number was accumulated as said diffusion error, and to memorize it.
[Claim 6] It is the image processing system which is a means are an image processing system according to claim 1, and said gradation error maintenance means is a means hold said gradation error about two or more pixels of said predetermined number, and compute the diffusion error to the specific pixel which said diffusion error storage means has in two or more of said non-judged pixels, and all said gradation errors currently held diffuse from two or more gradation errors currently this held, match with this specific pixel, and memorize.
[Claim 7] It is an image processing system according to claim 1. Said diffusion error storage means About each of the 1st non-judged pixel in the 1st [of said predetermined number / of two or more pixel circumference] predetermined field Said diffusion error is computed based on said gradation error about two or more pixels of this predetermined number. this -- about the 1st diffusion error storage means which matches with the 1st non-judged pixel and is memorized, and the 2nd non-judged pixel which exists on the outskirts of two or more pixels of said predetermined number, and is not into said 1st predetermined field whenever [by which said gradation error is computed] -- this gradation error -- being based -- this -- said diffusion error to each of the 2nd non-judged pixel -- computing -- this -- an image processing system equipped with the 2nd diffusion error storage means which matches with the 2nd non-judged pixel and is memorized.
[Claim 8] It is the image processing system which is a means to be an image processing system according to claim 7, and for said 2nd diffusion error storage means to compute said diffusion error about said 2nd non-judged pixel in the 2nd predetermined field which includes said 1st predetermined field, and to memorize.
[Claim 9] An image processing system equipped with the diffusion range means for switching which is an image processing system according to claim 8, and switches the range which makes said non-judged pixel diffuse said gradation error according to the conditions in connection with decision of said dot formation existence to said 1st predetermined field of two or more pixel circumference of said predetermined number, and said 2nd predetermined field.
[Claim 10] It is the image processing system which is the means which switches the range which diffuses said gradation error based on the gradation value of two or more pixels of said predetermined number, and the decision result of the dot formation existence about

this each pixel as conditions in connection with [are an image processing system according to claim 9, and] decision of said dot formation existence in said diffusion range means for switching.

[Claim 11] It is the image processing system which is a means to be an image processing system according to claim 1, and for said diffusion error storage means to compute said diffusion error based on the gradation error of the pixel which adjoins mutually as a two or more pixels gradation error of said predetermined number, to match with said non-judged pixel, and to memorize.

[Claim 12] In consideration of said diffusion error memorized by being an image processing system according to claim 11, and matching said dot formation decision means with said non-judged pixel, and said diffusion error from the pixel which adjoins this non-judged pixel, it is the image processing system which is a means to judge the dot formation existence about this non-judged pixel.

[Claim 13] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel It is the image processing system which changes this image data into the image data of the transcription by the existence of dot formation. A gradation error storage means to compute the gradation error generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, to match it with this decision pixel, and to memorize it, A gradation error maintenance means to hold said gradation error memorized by being around the view pixel which is going to judge the existence of said dot formation, and matching for every pixel judged [finishing / decision of this dot formation existence] for this every judged pixel, A dot formation decision means to judge dot formation existence based on the gradation value of said view pixel in said image data, taking into consideration the gradation error currently held for said every judged pixel, Said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel is detected. This gradation error that is not used for the decision about a view [degree] pixel is updated. the gradation error which read the gradation error of the judged pixel by which said gradation error is not temporarily held in the detected this judged pixel from said gradation error storage means, and carried out this reading appearance -- this -- An image processing system equipped with the renewal means of a gradation error with which decision of this following view pixel is presented.

[Claim 14] It is the image processing system which is an image processing system according to claim 13, and is a means to be equipped with a maintenance means temporarily hold the gradation error produced in said view pixel temporarily, and to update said gradation error which said renewal means of a gradation error is the gradation error held temporarily [said], and is not used for the decision about said view [degree] pixel.

[Claim 15] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel

It is the image-processing approach of changing this image data into the image data of the transcription by the existence of dot formation. The gradation error generated in the pixel this judged whenever it judged the existence of said dot formation Compute based on the decision result of this dot formation existence, and it holds temporarily. It is based on said gradation error about two or more pixels of a predetermined number held temporarily. The diffusion error diffused in the non-judged pixel by which decision of dot formation existence is not made is computed by being on the outskirts of two or more pixels of this predetermined number. The image-processing approach of judging the dot formation existence about this non-judged pixel based on the gradation value of this non-judged pixel in said image data, taking into consideration said diffusion error which matched said computed diffusion error with said non-judged pixel, memorized it, and was memorized by matching with said non-judged pixel.

[Claim 16] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The gradation error which is the image-processing approach of changing this image data into the image data of the transcription by the existence of dot formation, and is generated in the pixel this judged whenever it judged the existence of said dot formation is computed based on the decision result of this dot formation existence. Said gradation error which matches with this decision pixel, memorizes and is memorized by being around the view pixel which is going to judge the existence of said dot formation, and matching for every pixel judged [finishing / decision of this dot formation existence] Taking into consideration the gradation error which holds for this every judged pixel and is held for said every judged pixel Dot formation existence is judged based on the gradation value of said view pixel in said image data. Detect said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel, and the gradation error which is the judged pixel by which said gradation error is not held in said detected judged pixel is read. this gradation error that carried out reading appearance -- this -- after updating this gradation error that is not used for the decision about a view [degree] pixel -- this -- the image-processing approach with which decision of a view [degree] pixel is presented.

[Claim 17] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel With outputting these print data to the printing section which changes into the print data of the transcription by the existence of dot formation of this image data, forms an ink dot on print media, and prints an image A gradation error maintenance means to compute the gradation error which is the print control unit which controls this printing section, and is generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, and to hold it temporarily, It is based on said gradation error about the pixel of a predetermined number held temporarily. A diffusion error storage means to compute the diffusion error diffused in the non-judged pixel by which decision of dot formation existence is not made by being on the outskirts of two or more pixels of this predetermined number, to match the this computed diffusion

error with this non-judged pixel, and to memorize it, Taking into consideration said diffusion error memorized by matching with said non-judged pixel A dot formation decision means to judge the dot formation existence about this non-judged pixel based on the gradation value of this non-judged pixel in said image data, A print control unit equipped with a print-data output means to change said image data into said print data based on the decision result of said dot formation existence, and to output to said printing section.

[Claim 18] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel With outputting these print data to the printing section which changes into the print data of the transcription by the existence of dot formation of this image data, forms an ink dot on print media, and prints an image A gradation error storage means to be the print control unit which controls this printing section, to compute the gradation error generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, to match with this decision pixel, and to memorize, A gradation error maintenance means to hold said gradation error memorized by being around the view pixel which is going to judge the existence of said dot formation, and matching for every pixel judged [finishing / decision of this dot formation existence] for this every judged pixel, A dot formation decision means to judge dot formation existence based on the gradation value of said view pixel in said image data, taking into consideration the gradation error currently held for said every judged pixel, Said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel is detected. This gradation error that is not used for the decision about a view [degree] pixel is updated. the gradation error which read the gradation error which is the judged pixel by which said gradation error is not held in the detected this judged pixel from said gradation error storage means, and carried out this reading appearance -- this -- A print control unit equipped with a print-data output means to change said image data into said print data based on the decision result of the renewal means of a gradation error with which decision of this following view pixel is presented, and said dot formation existence, and to output to said printing section.

[Claim 19] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The program which realizes the approach of changing this image data into the image data of the transcription by the existence of dot formation The function to compute the gradation error which is the record medium recorded possible [reading] by computer, and is generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, and to hold it temporarily, The function which computes the diffusion error diffused in the non-judged pixel by which it is on the outskirts of two or more pixels of this predetermined number, and decision of dot formation existence is not made based on said gradation error about two or more pixels of a predetermined number held temporarily, The function to match said computed diffusion error with said non-judged pixel, and to memorize it, The record medium which recorded

the program which realizes the function to judge the dot formation existence about this non-judged pixel, based on the gradation value of this non-judged pixel in said image data, taking into consideration said diffusion error memorized by matching with said non-judged pixel.

[Claim 20] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The program which realizes the approach of changing this image data into the image data of the transcription by the existence of dot formation Are the record medium recorded possible [reading] by computer, and the gradation error generated in the pixel this judged whenever it judged the existence of said dot formation is computed based on the decision result of this dot formation existence. Said gradation error memorized by being around the view pixel which is going to judge the existence of said dot formation to be the function which matches with this decision pixel and is memorized, and matching for every pixel judged [finishing / decision of this dot formation existence] Considering the gradation error currently held for said every judged pixel as the function held for this every judged pixel The function to judge dot formation existence based on the gradation value of said view pixel shown in said image data, The function to detect said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel, The gradation error which is the judged pixel by which said gradation error is not held in said detected judged pixel is read. this gradation error that carried out reading appearance -- this -- after updating this gradation error that is not used for the decision about a view [degree] pixel -- this -- the record medium which recorded the program which realizes the function with which decision of a view [degree] pixel is presented.

[Claim 21] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The approach of changing this image data into the image data of the transcription by the existence of dot formation The function to compute the gradation error which is a program for realizing using a computer and is generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, and to hold it temporarily, The function which computes the diffusion error diffused in the non-judged pixel by which it is on the outskirts of two or more pixels of this predetermined number, and decision of dot formation existence is not made based on said gradation error about two or more pixels of a predetermined number held temporarily, The function to match said computed diffusion error with said non-judged pixel, and to memorize it, The program for realizing the function to judge the dot formation existence about this non-judged pixel, based on the gradation value of this non-judged pixel in said image data, taking into consideration said diffusion error memorized by matching with said non-judged pixel.

[Claim 22] The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The approach of changing this image data into the image data of the transcription by the

existence of dot formation The gradation error which is a program for realizing using a computer and is generated in the pixel this judged whenever it judged the existence of said dot formation is computed based on the decision result of this dot formation existence. Said gradation error memorized by being around the view pixel which is going to judge the existence of said dot formation to be the function which matches with this decision pixel and is memorized, and matching for every pixel judged [finishing / decision of this dot formation existence] Considering the gradation error currently held for said every judged pixel as the function held for this every judged pixel The function to judge dot formation existence based on the gradation value of said view pixel shown in said image data, The function to detect said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel, the gradation error which read the gradation error which is the judged pixel by which said gradation error is not held in said detected judged pixel, and carried out this reading appearance -- this -- after updating this gradation error that is not used for the decision about a view [degree] pixel -- this -- the program for realizing the function with which decision of a view [degree] pixel is presented.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the technique of changing this image data into the image data of the transcription by the formation existence of the dot about each pixel, in detail about the technique of changing the image data expressed with the gradation value of two or more pixels which constitute an image.

[0002]

[Description of the Prior Art] The image display device expressing an image is widely used as an output unit of various image devices by forming a dot on print media or a display medium called a liquid crystal screen. Although this image display device cannot express only one condition of whether a dot is formed or not, it is locally possible by controlling the formation consistency of a dot appropriately according to the gradation value of an image for gradation to express the image which changes continuously.

[0003] In these image display devices, as typical technique for judging the existence of dot formation about each pixel, there is technique called an error diffusion method or technique called this and the average error minimum method equivalent to a mathematics target so that a dot may be formed by the suitable consistency according to the gradation value of an image.

[0004] In the error diffusion method's diffusing and memorizing the error of the gradation expression produced having formed the dot in the view pixel, or by having not formed a dot to the non-judged pixel of the view pixel circumference, and judging the dot formation existence about a non-judged pixel, it is the technique of judging that dot formation

existence cancels the error diffused from the circumference pixel. Moreover, the average error minimum method is the technique of judging the dot formation existence about a view pixel, as it memorizes to the view pixel, without diffusing the error of the gradation expression produced by decision of dot formation existence in a circumference pixel, instead it faces judging the dot formation existence about a non-judged pixel, and the error memorized by the circumference pixel is read and these errors are negated. Also in which [these] technique, since it judges that the formation existence of a dot cancels the error of the gradation expression generated in the circumference pixel, a dot can be formed by the suitable consistency according to the gradation value of an image. Therefore, if the formation existence of a dot is judged with the application of such technique, it will become possible to display a high definition image with an image display device.

[0005]

[Problem(s) to be Solved by the Invention] However, since the formation existence of a dot was judged diffusing the error of the gradation expression generated in the view pixel in a surrounding non-judged pixel, only the part which makes these pixels diffuse and memorize an error had the problem that decision of dot formation existence will take time amount in the error diffusion method. Moreover, since the formation existence of a dot was judged reading the error of a gradation expression from a circumference pixel similarly about the average error minimum method, only the part which reads an error from a circumference pixel had the problem that decision of dot formation existence took time amount. If time amount is taken also in any decision of dot formation existence or case, it will become difficult to display an image quickly.

[0006] This invention is made in order to solve the above-mentioned technical problem in the conventional technique, and it aims at offering the technique which can be displayed quickly for a high-definition image by shortening the time amount which decision of dot formation existence takes, maintaining image quality equivalent to the case where an error diffusion method or the average error minimum method is applied.

[0007]

[The means for solving a technical problem, and its operation and effectiveness] The next configuration was used for the 1st image processing system of this invention in order to solve a part of above-mentioned technical problem [at least]. The image data which shows the gradation value of each pixel namely, by judging the existence of dot formation based on this gradation value for every reception and this pixel It is the image processing system which changes this image data into the image data of the transcription by the existence of dot formation. A gradation error maintenance means to compute the gradation error generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, and to hold it temporarily, It is based on said gradation error about two or more pixels of a predetermined number held temporarily. A diffusion error storage means to compute the diffusion error diffused in the non-judged pixel by which decision of dot formation existence is not made by being on the outskirts of two or more pixels of this predetermined number, to match the this computed

diffusion error with this non-judged pixel, and to memorize it, Let it be a summary to have a dot formation decision means to judge the dot formation existence about this non-judged pixel, based on the gradation value of this non-judged pixel in said image data, taking into consideration said diffusion error memorized by matching with said non-judged pixel.

[0008] Moreover, the 1st image-processing approach of this invention corresponding to the 1st above-mentioned image processing system The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel It is the image-processing approach of changing this image data into the image data of the transcription by the existence of dot formation. The gradation error generated in the pixel this judged whenever it judged the existence of said dot formation Compute based on the decision result of this dot formation existence, and it holds temporarily. It is based on said gradation error about two or more pixels of a predetermined number held temporarily. The diffusion error diffused in the non-judged pixel by which decision of dot formation existence is not made is computed by being on the outskirts of two or more pixels of this predetermined number. Let it be a summary to judge the DO@TTO formation existence about this non-judged pixel based on the gradation value of this non-judged pixel in said image data, taking into consideration said diffusion error which matched said computed diffusion error with said non-judged pixel, memorized it, and was memorized by matching with said non-judged pixel.

[0009] Said gradation error produced by judging the formation existence of a dot is held temporarily, and after computing the diffusion error to said non-judged pixel, it is made to memorize in this the 1st image processing system or image-processing approach based on the gradation error about two or more pixels of a predetermined number. If it carries out like this, since it can summarize the two or more pixels gradation error of a predetermined number to a surrounding non-judged pixel and it can be made to diffuse and memorize, it can be made to memorize quickly compared with the case where the gradation error produced in each pixel is made to diffuse and memorize according to an individual. Consequently, the time amount which decision of the existence of dot formation takes is shortened, and it becomes possible to display a high-definition image quickly.

[0010] In this 1st image processing system, said gradation error storage means is good also considering said gradation error as a storage means which can be written quickly rather than it writes to said diffusion error storage means.

[0011] Since the gradation error of said number of predetermined pixels is memorized by said diffusion error storage means once it is held at said gradation error storage means, this gradation error storage means is frequently written like this diffusion error storage means. Therefore, since R/W being quickly possible, then the time amount which time amount after this gradation error occurs until it finally memorizes as a diffusion error is shortened, as a result decision of the existence of dot formation takes can be shortened rather than it writes this gradation error storage means to this diffusion error storage means, it is suitable.

[0012] It may face matching said diffusion error with each sheep decision pixel, and

memorizing, and you may make it be a degree in this the 1st image processing system or conversion approach. Namely, the distribution error distributed to each of a non-judged pixel whenever said gradation error is computed is computed. When accumulating the computed this distribution error for every sheep decision pixel and accumulating said all distribution errors about two or more pixels of said predetermined number, it is good also as matching and accumulating these all distribution errors accumulated for every sheep decision pixel in said non-judged pixel as said diffusion error.

[0013] In this way, if the distribution error is accumulated in each sheep decision pixel whenever it searches for a gradation error, it can process quickly after computing said gradation error of said predetermined number until it matches said diffusion error with each sheep decision pixel and makes it memorize. All gradation errors can be made to diffuse and memorize by accumulating the diffusion error of several predetermined pixel minutes also about the non-judged pixel which cannot make all the gradation errors that should be diffused only by having diffused the gradation error for two or more pixels of a predetermined number once, and making it memorize memorize. Therefore, if the approach of starting is used, since processing which makes a non-judged pixel diffuse and memorize a diffusion error can be performed quickly, it becomes possible to shorten the time amount which decision of dot formation existence takes.

[0014] It may face matching a diffusion error with each sheep decision pixel, and memorizing, may change to an above-mentioned approach, and you may make it be a degree in this the 1st image processing system or conversion approach. That is, the diffusion error diffused in each sheep decision pixel based on the gradation error of the predetermined number which holds the gradation error about two or more pixels of a predetermined number, and is held is computed, and it is good also as matching and accumulating all the this computed diffusion errors in each sheep decision pixel.

[0015] In this way, since it becomes simple if two or more gradation errors of a predetermined number are held processing [which computes the diffusion error which should be diffused to each sheep decision pixel], as a result the processing which summarizes the gradation error of this number of predetermined pixels to a surrounding non-judged pixel, and diffuses it becomes simple, it is suitable. Moreover, since there are few these predetermined pixels if compared with this number of non-judged pixels, if the approach of starting is used, compared with the approach of accumulating a distribution error in each sheep decision pixel, whenever it computes a gradation error, storage capacity can be saved and it is desirable.

[0016] It may face matching a diffusion error with each sheep decision pixel, and memorizing, and you may make it be a degree in the 1st image processing system or conversion approach mentioned above. That is, the distribution error distributed to each of a non-judged pixel whenever said gradation error is computed is computed, and it is good also as matching the distribution error which is each sheep decision pixel in which the this computed distribution error is accumulated for every sheep decision pixel, and the distribution error for two or more pixels of said predetermined number was accumulated

with said non-judged pixel as said diffusion error, and memorizing it.

[0017] Thus, since it becomes possible to perform quickly processing which memorizing the distribution error which is the non-judged pixel in which the distribution error's was accumulated in each sheep decision pixel whenever it searched for the gradation error, and the distribution error of several predetermined pixel minutes was accumulated as said diffusion error, then the diffusion error of this number of predetermined pixels are matched [processing] with each sheep decision pixel, and makes them memorize, it is desirable.

[0018] It may face matching a diffusion error with a non-judged pixel and memorizing, may change to an above-mentioned approach, and you may make it be a degree in this the 1st image processing system or conversion approach. That is, it is good also as computing the diffusion error to the specific pixel which holds the gradation error about two or more pixels of a predetermined number, is in these two or more non-judged pixels, and all the gradation errors of this predetermined number diffuse from the gradation error currently this held, matching it with this specific pixel, and memorizing it.

[0019] In this way, the gradation error of the number of predetermined pixels is held, and if the diffusion error to said specific pixel is computed and it memorizes as said diffusion error, the processing which makes this diffusion error diffuse and memorize to a non-judged pixel will become simple. Consequently, since it becomes possible to shorten the time amount required in order for the processing which diffuses and memorizes the gradation error of this number of predetermined pixels to become quick, as a result to display an image, it is desirable.

[0020] It may face matching a diffusion error with a non-judged pixel and memorizing, and you may make it be a degree in the 1st image processing system or conversion approach mentioned above. namely, -- each of the 1st non-judged pixel in the 1st [of said predetermined number / of two or more pixel circumference] predetermined field -- said gradation error about two or more pixels of this predetermined number -- being based -- this -- said diffusion error to the 1st non-judged pixel -- computing -- this -- it matches with the 1st non-judged pixel and memorizes. whenever [moreover; / which computes said gradation error about the 2nd non-judged pixel which exists on the outskirts of two or more pixels of said predetermined number, and is not contained in said 1st predetermined field] -- from this gradation error -- this -- said diffusion error diffused to each of each 2nd sheep decision pixel -- computing -- this -- it matches with the 2nd non-judged pixel and memorizes.

[0021] If it carries out like this, since the distribution error distributed from two or more pixels of a predetermined number can be summarized to each sheep decision pixel and can be diffused about said 1st non-judged pixel, the time amount required in order to judge the existence of dot formation as a whole is shortened, and it is desirable.

[0022] In this image processing system, it is good also as computing and memorizing said diffusion error to the non-judged pixel in the 2nd predetermined field which included said 1st predetermined field as 2nd [said] non-judged pixel which a diffusion error is computed

whenever it computes a gradation error, and is diffused.

[0023] If it is going to make the large range diffuse the gradation error produced in two or more pixels of a predetermined number, since so many non-spread pixels will be contained, it is tended to complicate the processing for summarizing the gradation error of the number of predetermined pixels, and making it diffuse and memorize in the diffusion range of a gradation error. It divides into the 2nd predetermined field which includes the 1st predetermined field. on the other hand, the range which diffuses a gradation error -- the 1st predetermined field of the two or more pixels circumference of said predetermined number -- this -- this -- about the 1st non-judged pixel in the 1st predetermined field Whenever a gradation error is computed about the 2nd non-judged pixel which exists in addition to the 1st predetermined field, the diffusion error from this gradation error is computed. the gradation error about two or more pixels of a predetermined number -- being based -- the diffusion error to each pixel -- computing -- memorizing -- this -- this -- it matches and each of the 2nd non-judged pixel is made to memorize If it carries out like this, since the number of non-judged pixels which should diffuse the gradation error of the number of predetermined pixels collectively decreases, it becomes [to simplify processing] possible and is suitable. In addition, it faces diffusing the gradation error about two or more pixels of a predetermined number collectively to the 1st non-judged pixel in the 1st predetermined field, and making it memorize, and, of course, various kinds of approaches mentioned above can be applied suitably.

[0024] the range which makes a surrounding pixel diffuse said gradation error in this the 1st image processing system or conversion approach according to the conditions in connection with decision of said dot formation existence -- said 1st predetermined field of two or more pixel circumference of said predetermined number -- this -- you may make it switch to the 2nd predetermined field which includes the 1st predetermined field

[0025] The technique which switches extensive ** of the range which diffuses an error from the request on image quality etc. according to the conditions in connection with decision of dot formation existence may be used. in such a case, the range which diffuses a gradation error -- the 1st predetermined field of two or more pixel circumference of a predetermined number -- this, while switching to the 2nd predetermined field which includes the 1st predetermined field this -- since the processing which diffusing a gradation error directly, then the gradation error of the number of predetermined pixels are summarized [processing], and makes them diffuse and memorize about the non-judged pixel of the distant place which is not contained in the 1st predetermined field simplifies, it is suitable.

[0026] furthermore, the range which diffuses said error based on the gradation value of two or more pixels of said predetermined number, and the decision result of the dot formation existence about this each pixel -- said 1st predetermined field -- this -- it is good also as switching to the 2nd predetermined field which includes the 1st predetermined field.

[0027] For example, in the field where the gradation value of the image data which it is going to express is small, if the large range is made to diffuse the error by the dot having been formed when a dot is formed by chance, the dispersibility of a dot can be improved

and image quality can be raised. Or in the field where the gradation value of image data is big, if the large range is made to diffuse the error by a dot not having been formed when a dot is not formed by chance, image quality can be raised similarly. in such a case -- the case in which the dot was formed by chance while summarizing the diffusion error from two or more pixels of said predetermined number to the non-judged pixel of said 1st predetermined field and being spread where it is not case or formed -- this -- an error is diffused whenever it computes a gradation error also to the non-judged pixel of the distant place which is not contained to the 1st predetermined field. Without reducing most processing speed substantially, since it is rare to diffuse an error in a distant non-judged pixel, if it carries out like this, since the processing which judges dot formation existence can be simplified, it is suitable.

[0028] In the 1st image processing system or conversion approach mentioned above, it is good also as computing said diffusion error based on each gradation error about two or more pixels of this predetermined number as a pixel which adjoins mutually [said predetermined number] two or more pixels, matching with said non-judged pixel, and memorizing.

[0029] When [which the gradation error generated] two or more pixels adjoin mutually, compared with the case where it does not adjoin mutually, the part which overlaps mutually [the range which diffuses the gradation error generated in each pixel] becomes large. Therefore, since the non-judged pixel which the diffusion error from two or more gradation errors diffuses collectively will increase if it adjoins mutually [this predetermined number] two or more pixels when the diffusion error to each sheep decision pixel is computed from the gradation error of several predetermined pixel minutes and is diffused in each pixel, it becomes [diffusing a gradation error quickly and making it memorize] so possible and is suitable. In addition, it cannot be overemphasized that face diffusing the diffusion error which was computed from the gradation error of several predetermined pixel minutes in this case in a non-judged pixel, and making it memorize, and various kinds of approaches mentioned above can be applied suitably.

[0030] In the 1st above-mentioned image processing system or above-mentioned conversion approach of computing a diffusion error based on the two or more pixels gradation error which the predetermined number adjoined, it is good also as judging dot formation existence in consideration of said diffusion error from the pixel which adjoins this non-judged pixel in addition to the diffusion error memorized by facing judging the dot formation existence of a non-judged pixel, and already matching with this non-judged pixel.

[0031] Since the formation existence of a dot can be judged without performing processing which this non-judged pixel is made to diffuse and is made to memorize about the diffusion error diffused from the adjoining pixel when judging the formation existence of the dot about the non-judged pixel the dot formation existence of the adjoining pixel was judged to be, if it carries out like this, it becomes that judging quickly is possible and is suitable.

[0032] Moreover, the next configuration was used for the 2nd image processing system of this invention in order to solve a part of technical problem [at least] mentioned above. The

image data which shows the gradation value of each pixel namely, by judging the existence of dot formation based on this gradation value for every reception and this pixel It is the image processing system which changes this image data into the image data of the transcription by the existence of dot formation. A gradation error storage means to compute the gradation error generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, to match it with this decision pixel, and to memorize it, A gradation error maintenance means to hold said gradation error memorized by being around the view pixel which is going to judge the existence of said dot formation, and matching for every pixel judged [finishing / decision of this dot formation existence] for this every judged pixel, A dot formation decision means to judge dot formation existence based on the gradation value of said view pixel in said image data, taking into consideration the gradation error currently held for said every judged pixel, Said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel is detected. This gradation error that is not used for the decision about a view [degree] pixel is updated. the gradation error which read the gradation error of the judged pixel by which said gradation error is not temporarily held in the detected this judged pixel from said gradation error storage means, and carried out this reading appearance -- this -- Let it be a summary to have the renewal means of a gradation error with which decision of this following view pixel is presented.

[0033] The 2nd image-processing approach of this invention corresponding to the 2nd above-mentioned image processing system The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The gradation error which is the image-processing approach of changing this image data into the image data of the transcription by the existence of dot formation, and is generated in the pixel this judged whenever it judged the existence of said dot formation is computed based on the decision result of this dot formation existence. Said gradation error which matches with this decision pixel, memorizes and is memorized by being around the view pixel which is going to judge the existence of said dot formation, and matching for every pixel judged [finishing / decision of this dot formation existence] Taking into consideration the gradation error which holds for this every judged pixel and is held for said every judged pixel Dot formation existence is judged based on the gradation value of said view pixel in said image data. Detect said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel, and the gradation error which is the judged pixel by which said gradation error is not held in said detected judged pixel is read. this gradation error that carried out reading appearance -- this -- after updating this gradation error that is not used for the decision about a view [degree] pixel -- this -- let it be a summary to present decision of a view [degree] pixel.

[0034] In this the 2nd image processing system and image-processing approach, on the occasion of dot formation decision of a view pixel, the gradation error memorized by

matching with said judged pixel is held, and the formation existence of a dot is judged in consideration of the gradation error currently this held. In case the dot formation existence about a view [degree] pixel is judged, first, the judged pixel about this following view pixel is detected, and, subsequently difference with the judged pixel about this view pixel is investigated. And the gradation error memorized by the judged pixel which was not used for dot formation decision of this view pixel is read, and with the read gradation error, after updating the gradation error which is not used for decision of a view [degree] pixel, decision of this following view pixel is presented.

[0035] What is necessary is to read only the gradation error which is not used to dot formation decision about a view pixel, although it will be used for decision about this following view pixel on the occasion of dot formation decision of a view [degree] pixel if it carries out like this. Consequently, the number of gradation errors which must be read for dot formation decision can be decreased, and it becomes possible to shorten the time amount required in order to judge the formation existence of a dot.

[0036] It is the gradation error which holds the gradation error generated in said view pixel temporarily, and was this held, and you may make it update said gradation error which is not used for the decision about said view [degree] pixel in this the 2nd image processing system or conversion approach.

[0037] Since a dot formation judgment of a view [degree] pixel can be made without reading the gradation error memorized by matching with a view pixel if it carries out like this, it becomes possible to judge the formation existence of a dot quickly.

[0038] Moreover, in the print control unit which controls this printing section by outputting the print data for controlling formation of an ink dot to the printing section which forms an ink dot on print media and prints an image, the 1st image processing system or 2nd image processing system which this invention mentioned above can be used suitably. That is, in the 1st image processing system or 2nd image processing system mentioned above, it is quickly convertible for the image data according [accord / the image data which shows the gradation value of each pixel / reception] this image data to the formation existence of a dot. For this reason, if this the 1st image processing system or 2nd image processing system is applied to said print control unit, image data is quickly convertible for print data. In this way, in this printing section, if the obtained print data are outputted to said printing section, since it becomes possible to print a high definition image quickly, it is suitable.

[0039] Moreover, the program which realizes the 1st image-processing approach mentioned above or the 2nd image-processing approach is made to read into a computer, and this invention can also be realized using a computer. Therefore, this invention also contains the mode as following record media. Namely, the record medium corresponding to the 1st above-mentioned image-processing approach The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The program which realizes the approach of changing this image data into the image data of the transcription by the

existence of dot formation The function to compute the gradation error which is the record medium recorded possible [reading] by computer, and is generated in the pixel this judged whenever it judged the existence of said dot formation based on the decision result of this dot formation existence, and to hold it temporarily, The function which computes the diffusion error diffused in the non-judged pixel by which it is on the outskirts of two or more pixels of this predetermined number, and decision of dot formation existence is not made based on said gradation error about two or more pixels of a predetermined number held temporarily, The function to match said computed diffusion error with said non-judged pixel, and to memorize it, Let it be a summary to record the program which realizes the function to judge the dot formation existence about this non-judged pixel based on the gradation value of this non-judged pixel in said image data, taking into consideration said diffusion error memorized by matching with said non-judged pixel.

[0040] Moreover, the record medium corresponding to the 2nd above-mentioned image-processing approach The image data which shows the gradation value of each pixel by judging the existence of dot formation based on this gradation value for every reception and this pixel The program which realizes the approach of changing this image data into the image data of the transcription by the existence of dot formation Are the record medium recorded possible [reading] by computer, and the gradation error generated in the pixel this judged whenever it judged the existence of said dot formation is computed based on the decision result of this dot formation existence. Said gradation error memorized by being around the view pixel which is going to judge the existence of said dot formation to be the function which matches with this decision pixel and is memorized, and matching for every pixel judged [finishing / decision of this dot formation existence] Considering the gradation error currently held for said every judged pixel as the function held for this every judged pixel The function to judge dot formation existence based on the gradation value of said view pixel shown in said image data, The function to detect said judged pixel about the view [degree] pixel which judges dot formation existence to the degree of said view pixel, The gradation error which is the judged pixel by which said gradation error is not held in said detected judged pixel is read. this gradation error that carried out reading appearance -- this -- after updating this gradation error that is not used for the decision about a view [degree] pixel -- this -- let it be a summary to record the program which realizes the function with which decision of a view [degree] pixel is presented.

[0041] If the program currently recorded on these record media is made to read into a computer and various above-mentioned functions are realized using this computer, it will become possible to change quickly into the image data of the transcription by the existence of dot formation the image data which shows the gradation value for every pixel.

[0042]

[Embodiment of the Invention] In order to explain an operation and effectiveness of this invention more clearly, the gestalt of operation of this invention is explained below according to the following sequence.

A. In the C-1. 2nd example gestalt [of operation]: -- B. 1st example: -- B-1. equipment configuration: -- outline [of B-2. image data-conversion processing]: -- number transform-processing [of the B-3. 1st example] of gradation: -- B-4. modification: -- C. 2nd example: -- principle: which shortens the time amount of the number transform processing of gradation -- number transform-processing [of the C-2. 2nd example] of gradation: -- C-3. modification: -- D. 3rd example: -- E. 4th example: -- number transform-processing [of the principle:E-2. 4th example which shortens the time amount of the number transform processing of gradation in the E-1. 4th example] of gradation: [0043] A. The gestalt of operation : explain the gestalt of operation of this invention, referring to drawing 1 . Drawing 1 is an explanatory view for taking a printing system for an example and explaining the gestalt of operation of this invention. This printing system consists of the computers 10 and color printer 20 grades as an image processing system. A computer 10 will change this image data into the print data expressed by the formation existence of each color dot which can be printed by the color printer 20, if the gradation image data of a color picture is received from image devices, such as a digital camera and a color scanner. Conversion of this image data is performed using the program of the dedication called a printer driver 12. In addition, the gradation image data of a color picture can also be created by computer 10 using various application programs.

[0044] The printer driver 12 consists of two or more modules called the resolution conversion module, the color conversion module, the number conversion module of gradation, and interlace module. Processing which changes gradation image data into the transcription by the formation existence of a dot is performed by the number conversion module of gradation. About the processing performed by each of other module, it mentions later. A color printer 20 prints a color picture by forming each color ink dot on print media based on the print data changed by each [these] module.

[0045] The number conversion module of gradation in the printing system of this invention has the composition that the middle buffer was formed between the gradation error calculation section and an error buffer so that it may illustrate. It is possible to judge dot formation existence quickly by utilizing this middle buffer.

[0046] That is, generally, if the existence of dot formation of gradation data tends to express, in each pixel, the error of a gradation expression will occur naturally. The generated gradation error is accumulated in the mass error buffer prepared for a part of main storage area, and by the technique called the error diffusion method or the average error minimum method mentioned above, in case the dot formation existence about other pixels is judged, the error accumulated in this pixel is read from an error buffer, and the existence of dot formation is judged, taking the value of the read error into consideration. In order to judge the existence of dot formation so that clearly from this, it is necessary to write data frequently to an error buffer. Since a mass buffer is used for an error buffer and R/W of data takes a certain amount of time amount, if it writes frequently, only the part will require time amount.

[0047] On the other hand, by the number conversion module of gradation in the printing

system of this invention, the frequency which reading and the back write to an error buffer so that it may explain to a detail is decreased by utilizing the middle buffer of small capacity rather than an error buffer. Since the field of small capacity is written repeatedly, a middle buffer is easier to write at a high speed than an error buffer. For example, even if it is expected that it will be read from a cache in almost all cases if it is going to read data from a middle buffer between main memory and CPU in the computer system which has the cache memory which can be written at a high speed rather than main memory and it does not carry out special consideration, a middle buffer can be automatically written at a high speed rather than an error buffer.

[0048] Moreover, when allowances are in the register of CPU, it is possible to assign some middle buffers [at least] further to the register which can be written rather than cache memory at high speed, and improvement in the speed can be attained further. In order to assign some middle buffers to the register of CPU, in C, it is possible to specify clearly by the source code of processing software by using register declaration. Furthermore, when a complementary has some which have the function which assigns the register which is automatically to a variable with high operating frequency in the high compiler of optimization capacity and such a compiler is used, it will not specify clearly but some middle buffers will be assigned to the register of CPU also for **.

[0049] Thus, in the usual computer system, even if the middle buffer of the small capacity by which repeat R/W is carried out does not perform special consideration, it can be written at a high speed. Of course, also when processing not by software but by hardware etc., the direction which used the middle buffer of small capacity can realize easily the configuration which can be written at a high speed.

[0050] If the R/W frequency to the error buffer which R/W of data takes a certain amount of time amount decreases, it will become possible [the part] to judge dot formation existence quickly. Of course, since processing equivalent to an error diffusion method or the average error minimum method is logically performed even if it uses the middle buffer, image quality equivalent to these approaches is maintainable. Various modes exist in the concrete method of utilizing a middle buffer, in order to decrease the frequency written to an error buffer, and the mode of these various kinds is explained below using various examples.

[0051] B. 1st example: -- B-1. equipment configuration: -- drawing 2 is the explanatory view showing the configuration of the computer 100 as an image processing system of the 1st example. A computer 100 is a computer of the common knowledge constituted focusing on CPU102 by connecting ROM104, RAM106 of each other, etc. by bus 116. CPU102 consists of a computing element which actually processes, and two or more registers which hold the data under processing temporarily. The data currently held at the register can be processed at a high speed rather than the data memorized by RAM106 whether you are Haruka. Of course, the special storage element called cache memory may be utilized instead of a register. Although cache memory is not like a register, data can be written at a high speed rather than RAM106. Moreover, if cache memory is utilized, it will become easy

to treat a lot of data than the case where a register is used.

[0052] The video interface V-I/F112 grade for driving peripheral-device interface P-I/F108 for performing transfer of the disk controller DDC 109, the peripheral device, and data for reading the data of a flexible disk 124 or a compact disk 126 and CRT114 is connected to the computer 100. The color printer 200 mentioned later and the hard disk 118 grade are connected to P-I/F108. Moreover, it is also possible to print a digital camera 120 and the image captured with the digital camera 120 or the color scanner 122 when connecting the color scanner 122 grade to P-I/F108. Moreover, if it equips with Network Interface Card NIC 110, the data memorized by the store 310 which connected the computer 100 to the communication line 300, and was connected to the communication line are also acquirable.

[0053] Drawing 3 is the explanatory view showing the outline configuration of the color printer 200 of the 1st example. A color printer 200 is an ink jet printer which can form the dot of cyanogen, a Magenta, Hierro, and 4 color ink of black. Of course, in addition to the ink of these 4 color, the ink jet printer which can form the ink dot of a total of six colors including cyanogen (light cyanogen) ink with low color concentration and Magenta (light Magenta) ink with low color concentration can also be used. In addition, below, each of cyanogen ink, Magenta ink, Hierro ink, and black ink shall be called for short C ink, M ink, Y ink, and K ink by the case.

[0054] The color printer 200 consists of the device in which drive the print head 241 carried in carriage 240, and the regurgitation of ink and dot formation are performed, a device in which this carriage 240 is made to reciprocate to the shaft orientations of a platen 236 by the carriage motor 230, a device in which a print sheet P is conveyed by the paper feed motor 235, and a control circuit 260 that controls formation of a dot, migration of carriage 240, and conveyance of a print sheet so that it may illustrate.

[0055] Carriage 240 is equipped with the ink cartridge 242 which contains K ink, and the ink cartridge 243 which contains the various ink of C ink, M ink, and Y ink. If carriage 240 is equipped with an ink cartridge 242,243, each ink in a cartridge will be supplied to the head 244 for ink regurgitation for every color prepared in the inferior surface of tongue of a print head 241 thru/or 247 through introductory tubing which is not illustrated. In the head 244 for ink regurgitation for every color thru/or 247, they are 48 nozzles Nz. The nozzle train arranged in the fixed nozzle pitch k is established 1 set at a time.

[0056] The control circuit 260 consists of CPU261, ROM262, and RAM263 grade, and it carries out the regurgitation of the ink droplet from each nozzle to suitable timing based on the print data supplied from a computer 100 while it controls horizontal scanning and vertical scanning of carriage 240 by controlling actuation of the carriage motor 230 and the paper feed motor 235. In this way, a color printer 200 can print a color picture by forming the ink dot of each color in the suitable location under control of a control circuit 260, and on print media.

[0057] In addition, various approaches are applicable to the approach of carrying out the regurgitation of the ink droplet from the ink discharge head of each color. That is, the method which carries out the regurgitation of the ink using a piezo-electric element, the

approach of making generate a bubble (bubble) in an ink path at the heater arranged to the ink path, and carrying out the regurgitation of the ink droplet, etc. can be used. Furthermore, you may be the printer of the method which forms an ink dot on a print sheet using phenomena, such as hot printing, and the method which makes the toner powder of each color adhere on print media using static electricity instead of carrying out the regurgitation of the ink.

[0058] Moreover, it is also possible to use a possible printer [****] and the so-called variable dot impact printer for the magnitude of the ink dot formed on a print sheet using the approach of controlling the magnitude of the ink droplet which carries out the regurgitation from an ink discharge head, or controlling the number of the ink droplets which carry out two or more regurgitation of the detailed ink droplet at once, and carry out the regurgitation.

[0059] B-2. The outline of image data-conversion processing : drawing 4 is a flow chart which shows the flow of the processing which changes this image data into print data, when the computer 100 as an image processing system of the 1st example adds a predetermined image processing to the received image data. This processing is started when the operating system of a computer 100 starts a printer driver. Hereafter, according to drawing 4 , image data-conversion processing of the 1st example is explained briefly.

[0060] A printer driver will start first reading of the RGB color picture data which should be changed, if image data-conversion processing is started (step S100). Subsequently, the resolution of the incorporated image data is changed into resolution for a color printer 200 to print (step S102). When the resolution of color picture data is lower than print resolution, new data are generated between contiguity image data by performing linear interpolation, and when conversely higher than print resolution, the resolution of image data is changed into print resolution by thinning out data at a fixed rate.

[0061] In this way, conversion of resolution performs color transform processing of color picture data (step S104). Color transform processing is processing which changes the color picture data currently expressed by the combination of the gradation value of R, G, and B into the image data expressed by the combination of the gradation value of each color used by the color printers 200, such as C, M, Y, and K. Color transform processing can be quickly performed by referring to the table of the three dimension called a color translation table (LUT).

[0062] The number transform processing of gradation is started following color transform processing (step S106). The number transform processing of gradation is the following processings. The gradation data changed by color transform processing are expressed as data which have 256 gradation width of face for every color. On the other hand, in the color printer 200 of this example, only either condition of "a dot is formed" and "not forming a dot" cannot be taken. That is, the color printer 200 of this example cannot express only 2 gradation locally. Then, it is necessary to change the image data which has 256 gradation into the image data expressed with 2 gradation which can express a color printer 200. The processing which changes such a number of gradation is the number transform processing

of gradation. As mentioned above, it is possible by utilizing a middle buffer in this example to judge formation existence of a dot quickly. The back is explained to a detail about the number transform processing of gradation.

[0063] In this way, if the number transform processing of gradation is ended, a printer driver will start interlace processing (step S108). Interlace processing is processing which rearranges the image data changed into the format of expressing the formation existence of a dot into the sequence which should be transmitted to a color printer 200 while taking the formation sequence of a dot into consideration. A printer driver outputs the image data which performed interlace processing and was finally obtained to a color printer 200 as print data (step S110). A color printer 200 forms the ink dot of each color on print media according to print data. Consequently, the color picture corresponding to image data is printed on print media.

[0064] Below, in the number transform processing of gradation of the 1st example, by utilizing a middle buffer explains the processing which judges formation existence of a dot quickly.

[0065] B-3. The number transform processing of gradation of the 1st example : explain briefly how to judge formation existence of a dot in the so-called error diffusion method as preparation for explaining the principle which shortens the time amount which utilizes a middle buffer and decision of dot formation existence takes.

[0066] Drawing 5 is the explanatory view having shown notionally signs that the formation existence of a dot was judged in an error diffusion method, making a circumference pixel diffuse the gradation error generated in the pixel which judged the formation existence of a dot. In addition, below, the pixel to which its attention was paid in order to judge the formation existence of a dot is called a view pixel. As shown in drawing 5 (a), the gradation error E00 should occur between the gradation values in the image data of this view pixel as a result of having judged dot formation existence by the pixel (view pixel) which coincided with P00. The part which gave the slash all over drawing shows the pixel [finishing / decision of dot formation existence]. In an error diffusion method, the multiplication of the predetermined weighting factor (error diffusion coefficient) is carried out to the gradation error E00, and the acquired value is diffused in the non-judged pixel of the view pixel circumference. In addition, the suffix used below shall mean the following contents. For example, a suffix "00" shows a view pixel, a suffix "01" shows the pixel on the right of a view pixel, and a suffix "0 -1" shows a pixel on the left. A suffix "10" shows the pixel just under a view pixel, and a suffix "1 [-] 0" shows the pixel right above a view pixel. Moreover, a suffix "11" shall think that "01" was combined with the suffix "10", and shall show the pixel diagonally below to the right of a view pixel.

[0067] Drawing 6 is the explanatory view showing the example of a setting of the error diffusion coefficient used in case a gradation error is diffused. In addition, the pixel location where the slash is attached by drawing 6 shows the location of a view pixel. The matrix which displayed the error diffusion coefficient from a view pixel to such a circumference pixel is called an error diffusion matrix. For example, in the error diffusion

matrix of drawing 6 (a), "one fourth" is set to the right-hand of a view pixel as a value of the error diffusion coefficient K01. Therefore, when such an error diffusion matrix is used, one fourth of the errors of the gradation error generated in the view pixel will be distributed to a pixel on the right. Similarly, one fourth of the errors of the gradation error produced in the view pixel also in each pixel of the lower left of a view pixel, right under, and the lower right are distributed. An error diffusion matrix is not restricted to what is illustrated to drawing 6 , but range, an error diffusion coefficient, etc. which diffuse an error can set up various values, and in an actual error diffusion method, a suitable error diffusion matrix is suitably used so that good image quality may be acquired. In addition, in order to avoid complicated-ization of explanation, the following explanation explains in the illustrated error diffusion matrix as what uses a matrix with the narrowest diffusion range, i.e., the error diffusion matrix of drawing 6 (a).

[0068] As for the thing which uses the matrix of drawing 6 (a) as an error diffusion matrix, then the gradation error E00 produced in the view pixel P00 as shown in drawing 5 (a), the gradation error E00 is distributed to a total of four pixels of pixel P1-1 of the pixel P01 on the right and the lower left, the pixel P10 of right under, and the lower right pixel P11 every [4 / 1/], respectively. In this way, it is necessary to memorize the error (diffusion error) diffused in each pixel of the view pixel circumference in the condition of having dissociated for every pixel. Therefore, a diffusion error is memorized by mass RAM106 (refer to drawing 2) which can memorize the diffusion error about many pixels.

[0069] If the gradation error about a pixel P00 is diffused in a circumference pixel, decision of dot formation existence will be shortly started about the pixel P01 on the right. Drawing 5 (b) is the explanatory view showing notionally signs that the dot formation existence about the view pixel P01 is judged. On the occasion of dot formation decision, first, the diffusion error distributed and accumulated in the view pixel P01 from the circumference pixel is read, and the image data of the view pixel P01 is amended with the read diffusion error. As shown in drawing 5 (b), the error diffused according to the above-mentioned error diffusion matrix is accumulated in the view pixel P01 from four pixels of the circumference pixel [finishing / dot formation decision] P-10, i.e., a pixel, a pixel P-11, a pixel P-12, and a pixel P00. The formation existence of a dot is judged by reading this diffusion error from RAM106, amending the image data of the view pixel P01, and comparing the acquired correction value with a predetermined threshold. About the detail of decision, it mentions later. In this way, if the formation existence of a dot is judged about the view pixel P01, since the new gradation error E01 will occur in a pixel P01, this gradation error is diffused in a circumference pixel according to an error diffusion matrix. Thus, dot formation existence is judged in the error diffusion method, being spread in two or more pixels which exist on the outskirts each time whenever a gradation error occurs in a view pixel. For this reason, since it is necessary to write data frequently to RAM106, the time amount which decision of dot formation existence takes only that part also becomes long.

[0070] On the other hand, in the number transform processing of gradation of this example, the time amount which decision of dot formation existence takes is shortened by utilizing a

middle buffer. Drawing 7 is the explanatory view showing the principle which shortens the time amount which utilizes a middle buffer and decision takes. In this example, the register built in CPU102 is used as a middle buffer. As mentioned above, a register can be processed at a high speed compared with RAM106. Of course, equivalent processing may be substantially performed not using a register but using cache memory. Hereafter, the principle which shortens the time amount which decision of dot formation existence takes in the 1st example is explained, referring to drawing 7 .

[0071] Drawing 7 (a) shows signs that dot formation existence was judged about the view pixel P00. As a result of having made this judgment, the gradation error E00 has occurred in the view pixel. Six rectangles displayed on the right-hand side of drawing 7 (a) display typically six registers used as a middle buffer. On [of explanation] expedient, and in the following, R01, R02, R1-1, and R10, R11, R12 and agreement are attached and distinguished to each register. In the number transform processing of gradation of the 1st example, the gradation error E00 generated in the view pixel P00 is not diffused in direct RAM106, but is once stored in a register. That is, the value of a register R01 is updated with the error which should be distributed to the pixel P01 on the right of a view pixel. The error which should be distributed to a pixel P01 can be searched for by K01 and E00 according to an error diffusion matrix. Similarly, it is register R1-1. A value is updated with the error which should be distributed to pixel P1-1 at the lower left of the view pixel P00, and it updates, respectively with the error which should distribute the value of a register R11 to the lower right pixel P11 with the error which should distribute the value of a register R10 to the pixel P10 of right under. Each error can be searched for according to an error diffusion matrix by K1-1, E00 and K10, E00 and K11, and E00.

[0072] In this way, if the value of four corresponding registers is updated with each diffusion error which should be distributed to four surrounding pixels, the formation existence of a dot will be shortly judged about the pixel P01 on the right of a pixel P00. On the occasion of decision of a pixel P01, it is a value adding the error which read the diffusion error of the new view pixel P01 from RAM106, and read it, and the error memorized by the register R01, and the gradation data of the view pixel P01 are amended. Thus, if it amends with the value adding the diffusion error of the view pixel P01, and the error of a register R01, the processing same substantial completely can be performed with the usual error diffusion method. That is, as explained using drawing 5 , in case this is diffused in the surrounding pixel whenever a gradation error occurs and the dot formation existence of the pixel P01 on the right is judged in the usual error diffusion method, the diffusion error distributed to the pixel P01 is read, and the formation existence of a dot is judged based on the value which amended and amended the gradation data of a pixel P01. On the other hand, in this example, since the error distributed to a pixel P01 is memorized by the register R01 as shown in drawing 7 (a), it is a value adding the diffusion error of a pixel P01, and the error memorized by the register R01, and processing equivalent to the usual error diffusion method substantial completely can be performed by amending gradation data. In this way, if the formation existence of a dot is judged about the new view

pixel P01, the new gradation error E01 will occur in the view pixel P01 (refer to drawing 7 (b)).

[0073] Each register is made to diffuse the gradation error E01 generated in the pixel P01 as well as the gradation error E00 of a pixel P00 at a rate set as the error diffusion matrix. It explains concretely, referring to drawing 7 (b). First, since a pixel P02 is a pixel on the right of the view pixel P01, it updates the error diffusion coefficient K01 set as the error diffusion matrix in the value of the corresponding register R02, and the gradation error E01 with the value which carried out multiplication. value of the register R10 which corresponds since a pixel P10 is a pixel at the lower left of the view pixel P02 **** -- error diffusion coefficient K1-1 Value K1-1 which carried out the multiplication of the gradation error E01, and E01 are added. Here, as shown in drawing 7 (b), since the error K10 already diffused from the pixel P00 and E00 are memorized by the register R10, the already diffused error and the newly diffused error are added to it on a register R02. Consequently, the value of a register R02 is updated by the sum (K10 and E00+K1-1, and E01) of the error diffused from two pixels. The value of the register R11 which corresponds similarly about a pixel P11 and a pixel P12 is hereafter updated with a diffusion error (K11 and E00+K10, E01), and the value of a register R12 is updated by K11 and E01.

[0074] It means that the gradation error E00 generated in the pixel P00 and the gradation error E01 in a pixel P01 were spread in each register by performing the above processings. Then, as shown in drawing 7 (c), the error memorized by each register is written in the error buffer (specifically RAM106) corresponding to a circumference pixel. Namely, the value K01 memorized by the register R02 and E01 are added to the error buffer of a pixel P02. Pixel P1-1 In an error buffer, it is register R1-1. Value K1-1 memorized and E00 Value K11 and E00+K10 of a register R11, and E01 are added to the pixel P11, and the value K11 of a register R12 and E01 are added for the value (K10 and E00+K1-1, and E01) of a register R10 to the pixel P12 at the pixel P10, respectively. By carrying out like this, the same error as the condition (refer to drawing 5 (c)) of having diffused the error in a pixel P00 and a pixel P01 in the usual error diffusion method will be spread in each error buffer of five surrounding pixels. In addition, since decision of dot formation existence is already ended about the pixel P01, it is not necessary to add the value of a register R01 to RAM106. Thus, by the approach of this example, since the dot formation existence for 2 pixels was judged and the error is diffused in the error buffer for 5 pixels, whenever it judges 1 pixel, error diffusion to an error buffer will be performed at a rate of 2.5 pixels.

[0075] In addition, above, all the errors diffused in five pixels of the pixel circumference of explanation which performed binary-ization for convenience were once memorized to the register, and after adding on the register the error diffused from two pixels, it explained as what is diffused in an error buffer. Of course, about the pixel which does not add an error on a register, pixel P1-1 [for example,], a pixel P02, and pixel P12, an error may be directly diffused in an error buffer.

[0076] In the usual error diffusion method shown in drawing 5 , whenever it made a 1-pixel dot formation judgment, the error was diffused in 4 pixels of the circumference. On the

other hand, what is necessary is just to diffuse an error in the error buffer for 2.5 pixels in the approach of this example, whenever it judges 1 pixel by being spread collectively, once it adds on a register the error diffused from two or more pixels to the same pixel which has not yet been made binary. Of course, although the process which distributes an error to a register is added to the usual error diffusion method, since processings, such as renewal of the value of a register or addition on a register, can be performed for whether being Haruka at a high speed compared with the processing which diffuses an error to RAM, it becomes possible to shorten the time amount which decision of dot formation existence takes as a whole. In addition, although explained as what is memorized to a register here, it must be the storage means which can be written at a high speed rather than the storage means which is not necessarily restricted to a register but is used as an error buffer. Moreover, for example, in order to exchange the data of CPU and RAM at high speed, equivalent processing may be substantially performed by utilizing the cache memory prepared. In order that the middle buffer used by this example may repeat and use the memory area of small capacity for processing of each pixel, even if a designer does not usually specify clearly, it is automatically assigned to a register and cache memory by the function of a compiler or the CPU itself, and becomes possible [writing at a high speed].

[0077] In addition, in the above explanation, although the error diffused from two pixels shall be directly added on a register, once memorizing each error to a separate register, you may add on another register. For example, if it explains taking the case of the error diffused in the pixel P10 of drawing 7 R> 7 (b), the value of a register R10 will be first updated by the error K10 from a pixel P00, and E00, and, subsequently error K1-1 from a pixel P01 and E01 will be added on a register R10. It changes to such processing and is Register Ra, respectively about the error from a pixel P00, and a pixel P01 - an error. Register Rb It memorizes and, of course, the value of two registers may be added on a register R10.

[0078] Drawing 8 is a flow chart which shows the flow of the number transform processing of gradation of this example explained above. This processing is performed by CPU102 of a computer 100. In addition, below, as mentioned above, the color printer of this example is explained, without specifying the color of an ink dot, in order to avoid complicated-ization of explanation, although it is the printer which can form the ink dot of four colors of C, M, Y, and K and the number transform processing of gradation shown in drawing 8 is also performed for every color. In addition, in addition to the four above-mentioned colors, LC ink and LM ink may be added and, of course, 6 color printer may be used. [0079] Moreover, as mentioned above, the color printer of this example can also be considered as the variable dot impact printer which can form the dot from which magnitude differs for every color. When using a variable dot impact printer (for example, when using the variable dot impact printer which can form the various dots of a large dot, an inside dot, and a small dot), the number transform processing of gradation explained below is performed for every dot of various magnitude.

[0080] Thus, since the count which performs the number transform processing of gradation

increases as the color of the ink to be used increases or formation of the dot of various magnitude is attained, there is an inclination for the time amount which processing takes so much to also become long. Since quick processing is possible, the number transform processing of gradation of this example explained below is suitably [in such a case] applicable.

[0081] Initiation of the number transform processing of gradation of this example reads first the image data Cd1 and Cd2 for 2 pixels which are going to judge dot formation existence (step S200). In addition, a left-hand side pixel is made for convenience to call the 2nd pixel the 1st pixel and the pixel of a call and right-hand side here. Image data is memorized by RAM106 built in the computer 100. Subsequently, the diffusion errors Ed1 and Ed2 accumulated in each (the 1st pixel and the 2nd pixel) by being spread are read from an error buffer (step S202). The error buffer is also formed on RAM106.

[0082] Then, the pixel [1st] amendment data Cx1 are computed by adding the pixel [1st] image data Cd1 and the pixel [1st] diffusion error Ed1 (step S204). In this way, the obtained amendment data Cx1 and the predetermined threshold th Value [meaning forming a dot] "1" Compare (step S206), if the amendment data are larger, it will judge that a dot is formed in the 1st pixel, and write in the variable Cr1 which shows the decision result about the 1st pixel (step S208). Otherwise, it judges that a dot is not formed in the 1st pixel, and the value "0" meaning a dot not forming in a variable Cr1 is written in (step S210).

[0083] In this way, if the dot formation existence about the 1st pixel is judged, the gradation error generated in the 1st pixel with this decision will be computed (step S212). Gradation error E1 generated in the 1st pixel It can ask by subtracting the gradation value (such a gradation value being called result value below) expressed by the 1st pixel forming a dot or by not forming a dot from the pixel [1st] amendment data Cx1.

[0084] Subsequently, gradation error E1 generated in the 1st pixel It is spread in a middle buffer (step S214). This processing is processing equivalent to the processing explained using drawing 7 (a). Namely, pixel [1st] gradation error E1 The multiplication of the error diffusion coefficient set as the error diffusion matrix is carried out, the diffusion error which should be diffused in a circumference pixel is computed, and the acquired diffusion error is memorized to each middle buffer.

[0085] If the error generated in the 1st pixel is diffused in a middle buffer, decision of the dot formation existence about the 2nd pixel will be started. In order to make a judgment about the 2nd pixel, it is necessary to compute pixel [2nd] amendment data. Then, the pixel [2nd] amendment data Cx2 are computed by adding the pixel [2nd] image data Cd2 read beforehand, the pixel [2nd] diffusion error Ed2, and the error distributed to the 2nd pixel from the 1st pixel (step S216). As explained using drawing 7 , the error diffused in the 2nd pixel from the 1st pixel is memorized by the middle buffer defined beforehand. In addition, in the example shown in drawing 8 , pixel [2nd] image data and a diffusion error shall be beforehand read at the same time it reads pixel [1st] image data and a diffusion error, and, of course, after it is needed for calculation of amendment data, it is good about

the 2nd pixel, also as what reads image data and a diffusion error. In reading the image data about the 1st pixel and the 2nd pixel, and a diffusion error at once, it becomes possible to read these data quickly by adopting the approach of reading from the location which continued on memory.

[0086] In addition, you may make it read about pixel [2nd] image data and a diffusion error just before step S216 instead of steps S200 and S202. If it carries out like this, when a limitation is in the number of registers of CPU102, since it becomes possible [using register for other purposes] in between [until it reads pixel / 2nd / image data and a diffusion pixel], it is suitable.

[0087] If the amendment data Cd2 about the 2nd pixel are computed at step S216, it is the predetermined threshold th like the case of the 1st pixel. The formation existence of a dot is judged by comparing (step S218). The value "0" meaning judging that a dot is formed in the 2nd pixel, if the amendment data Cd2 are larger, writing in the value "1" meaning forming a dot (step S220), otherwise, not forming a dot is written in (step S222). Subsequently, pixel [2nd] gradation error E2 It computes (step S224). Gradation error E2 generated in the 2nd pixel Pixel [1st] gradation error E1 It can ask from the pixel [2nd] amendment data Cx2 similarly by subtracting a value (gradation value expressed by the pixel by having judged the formation existence of the dot about a pixel) the result about the 2nd pixel.

[0088] Then, processing which diffuses a pixel [2nd] gradation error in a middle buffer is performed (step S226). This processing is processing equivalent to the processing explained using drawing 7 (b). Namely, pixel [2nd] gradation error E2 The multiplication of the error diffusion coefficient of each pixel set as the error diffusion matrix is carried out, the diffusion error which should be diffused in a circumference pixel is computed, and the acquired diffusion error is stored in each middle buffer. Here, when the error distributed to the middle buffer from the 1st pixel is already memorized, the newly distributed error is added and memorized for the already memorized error. Since the error already distributed about two registers, a register R10 and a register R11, exists, this is made to add and memorize the newly distributed error in the example shown in drawing 7 (b). Consequently, the error which should be diffused in each surrounding pixel will be accumulated in five registers from the 1st pixel and the 2nd pixel. So, in processing of continuing step S228, if the error memorized for every pixel is added to the error buffer on RAM (refer to drawing 7 (c)) and it finishes adding it to a middle buffer, the middle buffer will be reset.

[0089] Subsequently, it judges whether dot formation existence was judged about all pixels (step S230), and if the non-judged pixel remains, a series of processings which return and follow step S200 will be performed. In this way, if it judges 2 pixels of formation existence of a dot at a time and decision is ended about all pixels, diffusing a gradation error in a circumference pixel until it ends decision of dot formation existence about all pixels, the number transform processing of gradation of this example will be ended, and it will return to image data-conversion processing of drawing 4 .

[0090] What is necessary is just to diffuse an error in the error buffer for 5 pixels in the

number transform processing of gradation of this example explained above, whenever it judges the dot formation existence for 2 pixels. If it compares with an error being diffused in the error buffer for 4 pixels whenever it makes a 1-pixel judgment in order to perform same processing using the usual error diffusion method, the time amount which diffusion of the error to an error buffer takes can be greatly shortened by adopting the number transform processing of gradation of this example. Since the generated gradation error must be diffused in a circumference pixel while judging dot formation existence especially about the pixel of a large number which constitute an image from number transform processing of gradation, the time amount which diffusion of the error to an error buffer takes has accounted for the comparatively big rate in the time amount which the number transform processing of gradation takes. If the time amount which adopts the number transform processing of gradation of this example, and diffusion with error takes from this is shortened, it will become possible to be able to perform quickly the whole image processing including the number transform processing of gradation, as a result to print an image quickly.

[0091] The above explanation explained for convenience as a thing of explanation which judges 2 pixels of formation existence of a dot at a time. That is, the gradation error produced in two pixels is stored in the middle buffer, and the error diffused from two pixels is collectively added and diffused in an error buffer on a middle buffer. Of course, dot formation existence is judged about more pixels, and you may make it add the generated error on a middle buffer. For example, what is necessary is just to diffuse an error in the error buffer of six pixels which are around the pixel (a slash is attached and displayed all over drawing) which judges dot formation existence, as shown in drawing 9 (a) when judging 3 pixels of formation existence of a dot at a time. That is, it is sufficient if an error is diffused in the error buffer for 2 pixels per [which judges] pixel. Furthermore, what is necessary is just to diffuse an error in the error buffer for 1.5 pixels per decision pixel, since what is necessary is just to diffuse an error in the error buffer for 9 pixels as shown in drawing 9 (b) when it judges 6 pixels of formation existence of a dot at a time. Thus, since the number of pixels of the error buffer diffused in per 1 pixel of decision pixels decreases so that the number of pixels which judges dot formation existence at once is made [many], it becomes possible to shorten the time amount which the number transform processing of gradation takes.

[0092] B-4. Modification : although explained as what computes the diffusion error diffused in a circumference pixel from the 1st pixel and the 2nd pixel in the number conversion approach of gradation of the 1st example explained above based on each gradation error, and is accumulated for every circumference pixel, even if as follows, it is good again. That is, the diffusion error which should be diffused in a circumference pixel from the gradation error which memorized the gradation error itself generated in the 1st pixel and the 2nd pixel to the middle buffer, and has been these-memorized is computed, and you may make it make the error buffer corresponding to each pixel memorize. Although this approach has the difference in whether it is the diffusion error which the value memorized by the middle

buffer is accumulating, or it is a gradation error, it can perform equivalent processing to an error buffer substantially from the same error being memorized. Hereafter, the modification of such 1st example is explained.

[0093] Drawing 10 is a flow chart which shows the flow of the number conversion approach of gradation in the modification of the 1st example. The number conversion approach of gradation of a modification is briefly explained focusing on a hereafter different part from the number transform processing of gradation of the 1st example shown in drawing 8 , referring to drawing 10 R> 0.

[0094] If processing as well as the number transform processing of gradation of the 1st example is started also in a modification, image data (the 1st pixel and the 2nd pixel) Cd1 and Cd2 will be read first (step S300). Here, the pixel of the left-hand side in two pixels which are going to judge dot formation existence is said, and a right-hand side pixel is said [pixel / 1st] in the 2nd pixel. Subsequently, the diffusion errors Ed1 and Ed2 diffused in each (the 1st pixel and the 2nd pixel) are read from an error buffer (step S302). The error buffer is formed on RAM106 like the case of the 1st example. Subsequently, the pixel [1st] amendment data Cx1 are computed by adding pixel [1st] image data and a diffusion error (step S304). Then, the amendment data Cx1 are compared with the predetermined threshold th (step S306), if the amendment data are larger, it will judge that a dot is formed in the 1st pixel, and the value "1" meaning forming a dot is written in the variable Cr1 which shows the decision result about the 1st pixel (step S308). Otherwise, the value "0" which judges that a dot is not formed in the 1st pixel and means not carrying out dot formation is written in a variable Cr1 (step S310). In this way, if the existence of the dot formation about the 1st pixel is judged, the gradation error generated in the 1st pixel with this decision will be computed (step S312). A gradation error can be searched for by subtracting a value from the amendment data Cx1 a result like the case of the 1st example.

[0095] Pixel [1st] gradation error E1 computed in this way in the modification of the 1st example It memorizes to a middle buffer (step S314). That is, although the middle buffer was made to diffuse a gradation error in the number transform processing of gradation of the 1st example, it memorizes to a middle buffer as it is, without diffusing a gradation error in the number transform processing of gradation of a modification.

[0096] Pixel [1st] gradation error E1 If it memorizes to a middle buffer, in order to judge the dot formation existence about the 2nd pixel, the pixel [2nd] amendment data Cx2 are computed (step S316). That is, the pixel [2nd] image data Cd2 and the diffusion error Ed2 which were read beforehand, and the error diffused from the 1st pixel are added. The error from the 1st pixel is the pixel [1st] gradation error E1 memorized by the middle buffer. It asks by carrying out the multiplication of the error diffusion coefficient. The value of an error diffusion coefficient is set as the error diffusion matrix like the case of the 1st example. In this way, the computed pixel [2nd] amendment data Cx2 are compared with the predetermined threshold th (step S318), if the amendment data Cx2 are larger, it will judge that a dot is formed, and the value "1" meaning forming a dot is written in the variable Cr2 which shows the decision result about the 2nd pixel (step S320). Otherwise, it

judges that a dot does not form and the value "0" meaning not forming a dot in a variable Cr2 is written in (step S322). Subsequently, gradation error E2 about the 2nd pixel Gradation error E2 computed and (step S324) computed It once memorizes to a middle buffer as it is, without diffusing a circumference pixel (step S326). At this time, it is the pixel [2nd] gradation error E2. Pixel [1st] gradation error E1 memorized previously It memorizes separately. Pixel [2nd] gradation error E2 Pixel [1st] gradation error E1 A value can be similarly subtracted and calculated a result from the pixel [2nd] amendment data Cx2.

[0097] In this way, pixel [1st] gradation error E1 Pixel [2nd] gradation error E2 If it memorizes to each middle buffer, two gradation errors will be summarized and an error buffer will be diffused. This processing is explained using drawing 7 . In drawing 7 , a pixel P00 is equivalent to the 1st pixel, and the pixel P01 is equivalent to the 2nd pixel. As mentioned above, the gradation error generated in the 1st pixel and the 2nd pixel is diffused at a predetermined rate in five pixels around these pixels. pixel P1-1 [for example,] in the pixel [1st] lower left **** -- error K1-1 from a pixel P00 and E00 are spread. Here, it is K1-1. It is the error diffusion coefficient used in case a gradation error is diffused to a lower left pixel, and the value of an error diffusion coefficient is set as the error diffusion matrix. Moreover, in a pixel P10, error K10 and E00+K1-1 adding the error K10 from the 1st pixel, E00, and error K1-1 from the 2nd pixel and E01, and E01 are spread. Similarly about other three pixels, K11 and E01 are spread in a pixel P12, and the value of K01 and E01 is spread by K11 and E00+K10, and E01 in a pixel P02 at a pixel P11, respectively. Thus, the value of the error diffused in each surrounding pixel is the gradation error E1. And gradation error E2 If decided, based on an error diffusion coefficient, it is computable. Then, pixel [1st] gradation error E1 memorized by the middle buffer in step S328 Pixel [2nd] gradation error E2 The error diffused in each pixel of the circumference (the 1st pixel and the 2nd pixel) is computed by using, and it adds to the error buffer corresponding to each pixel. If such processing is completed, in order to use it for the processing which is the following pixel, it is the gradation error E1 and E2. The memorized middle buffer is reset.

[0098] Subsequently, it judges whether it is the no which judged dot formation existence about all pixels (step S330), and if the non-judged pixel remains, a series of processings which return and follow step S300 will be performed. In this way, if the existence of dot formation is judged about all pixels, the number transform processing of gradation of the modification of the 1st example will be ended, and it will return to image data-conversion processing of drawing 4 .

[0099] Also by the approach of a modification explained above, equivalent processing can be substantially performed with the number transform processing of gradation of the 1st example. According to the approach of this modification, the capacity of a middle buffer can be saved rather than the approach of the 1st example.

[0100] The 2nd example : C. In the number transform processing of gradation of the 1st example explained more than principle: which shortens the time amount of the number

transform processing of gradation in the C-1. 2nd example The gradation error which judged dot formation existence and was generated is diffused in middle buffers, such as a register which can be written at a high speed, and all the errors diffused in the middle buffer whenever it ends decision of the dot formation existence of the number of predetermined pixels are added to the error buffer. On the other hand, in the number transform processing of gradation of the 2nd example explained below, whenever it judges dot formation existence, every 1 pixel is added to an error buffer, and it goes. According to the approach of the 2nd example, the number transform processing of gradation can be realized with the number of registers smaller than the approach of the 1st example, and the write-in frequency from middle buffers, such as a register, to an error buffer can also be further lessened so that it may mention later. Hereafter, this 2nd example is explained.

[0101] It explains referring to drawing 11 for the usual error diffusion method and the principle which performs equivalent processing substantially by adding the diffusion error for 1 pixel to an error buffer, whenever it judges the processing principle of the 2nd example of introduction, i.e., 1-pixel dot formation existence, and going. The part which attached the slash of drawing 11 shows the field of the pixel which made a dot formation judgment. In addition, an error diffusion matrix shall use the matrix shown in drawing 6 (a) like [the following] the case of the 1st example because of the facilities of explanation. Drawing 11 (a) shows signs that judged dot formation existence about a pixel P00, and the produced gradation error E00 is diffused in the circumference pixel.

[0102] Here, if its attention is paid to the pixel P11 enclosed and shown with the thick broken line, in addition to the error E00 produced in the pixel P00, the error E01 produced in the pixel P01 on the right of a pixel P00 and its error E02 further produced in the pixel P02 on the right are distributed to the pixel P11 (refer to drawing 11 (b) and (c)). That is, the error will be continuously distributed to one pixel from three continuous pixels. That in addition, an error is distributed from three continuous pixels By the case where the error diffusion matrix shown in drawing 6 (a) or drawing 6 (e) is used The error from seven pixels which it follows when the error from five pixels which continue when other matrices, for example, the matrix of drawing 6 (b) and drawing 6 (f), are used uses the matrix of drawing 6 R> 6 (c) will be distributed continuously, respectively. Thus, since an error is continuously distributed from the pixel which follows each pixel, if the error distributed is accumulated on a middle buffer for every pixel and it finishes accumulating the error of the number of predetermined pixels, the pixel is considered that diffusion with error was completed for a while, and the error accumulated in the middle buffer is written in the error buffer. In this way, if the error of the middle buffer considered that are recording with error was completed for a while is written in the corresponding error buffer of a pixel, since it becomes unnecessary to carry out, the frequency which writes an error in the error buffer of the same pixel repeatedly and which is written to an error buffer can be reduced.

[0103] Drawing 12 is the explanatory view having shown notionally signs that utilized the middle buffer and the principle mentioned above was realized. Drawing 12 (a) shows the condition of having judged dot formation existence by the pixel P00. Four rectangles shown

in the right-hand side of drawing 12 (a) show the register as a middle buffer typically, respectively. Below, it is R1, R2, R3, and R4 to each register. A sign is attached and identified. The multiplication of the error diffusion coefficient set as the error diffusion matrix is carried out, and it diffuses the gradation error E00 generated in the view pixel P00 in each register. In addition, in order to avoid that illustration makes it complicated, in drawing 12 , from a pixel P00, the error diffused in a lower left pixel is abbreviated to E00_LD, and is displayed. The value of error E00_LD is error diffusion coefficient K1-1 to the gradation error E00 of a pixel P00. It can ask by carrying out multiplication. Similarly, it is E00_D about the error diffused in the pixel of right under from a pixel P00. It is E00_R about the error which diffuses the error which omits, displays and is diffused in a lower right pixel from a pixel P00 in a right pixel from a pixel P00 with E00_RD. Suppose that it displays for short. the error which abbreviates the error diffused in a lower left pixel to E01_LD, displays it from a pixel P01 similarly about the gradation error generated in other pixels, and is diffused in the pixel of right under from a pixel P01 -- E01_D The error which diffuses the error diffused in a lower right pixel in E01_RD and a right pixel is E01_R. It omits, respectively and displays. Moreover, the asterisk given to the right-hand side of each register shows the register which finished accumulating the error of the number of predetermined pixels. About the contents which an asterisk means, it mentions later.

[0104] With reference to drawing 12 (a), the gradation error E00 produced in the view pixel P00 is a register R1 thru/or a register R4. Signs that it is accumulated in four registers are explained. The inside of four registers, and register R1 It is the register with which the error diffused in a right pixel from a view pixel is overwritten. Unlike other three registers, the error to the pixel on the right of a view pixel is always overwritten. drawing 12 (a) -- register R1 **** -- error E00_R diffused in the right pixel P01 from the view pixel P00 The value is overwritten. Each error diffused in each pixel of the lower left, right under, and the lower right from a view pixel is added to other three registers. Whether which error is added to which register changes in order with migration of a view pixel so that it may explain below.

[0105] first -- the case (refer to drawing 1212 (a)) where a view pixel is in a pixel P00 -- register R2 **** -- the error (error diffused in the pixel of right under from a view pixel) from a view pixel to a pixel P10 is added. register R3 **** -- a view pixel to pixel P1-1 An error (error diffused in a lower left pixel from a view pixel) is added. register R4 **** -- the error (error diffused in a lower right pixel from a view pixel) from a view pixel to a pixel P11 is added.

[0106] Next, the case where a view pixel moves to a pixel P01 is considered (refer to drawing 12 (b)). even if a view pixel moves -- register R2 **** -- the error to a pixel P10 is still added. since the pixel P10 is a pixel at the lower left of a view pixel, if after migration of a view pixel is said by physical relationship with a view pixel -- register R2 **** -- the error from a view pixel to a lower left pixel will be added. the time of a view pixel being in a pixel P00, if it puts in another way -- register R2 **** -- when a view pixel moves to a pixel

P01 to the error from a view pixel to the pixel of right under having been added, the error from a view pixel to a lower left pixel will be added.

[0107] the same -- register R4 **** -- although the error from a view pixel to a lower right pixel is added when a view pixel is in a pixel P00 since the error to a pixel P11 is added, when a view pixel moves to a pixel P01, the error from a view pixel to the pixel of right under will be added.

[0108] Register R3 If it attaches, when a view pixel is in a pixel P00, it is pixel P1-1 at the lower left of a view pixel. Although the error was added, after a view pixel moves to a pixel P01, it is already pixel P1-1. Diffusion with error is unnecessary. then, register R3 **** -- the error from the view pixel P02 to the new pixel P12 is added. It is a register R3 after all. Even if it attaches, when a view pixel moves, the error to a direction which sees and is different from a view pixel will be added like other registers. namely, -- the time of a view pixel being in a pixel P00 -- register R3 **** -- although the error from a view pixel to a lower left pixel is added, when a view pixel moves to a pixel P01, the error from a view pixel to a lower right pixel will be added.

[0109] Furthermore, it is a register R2 similarly [when it progresses and a view pixel moves to a pixel P02 from a pixel P01]. Or register R4 The error to a pixel which tends to see and be different from a view pixel will be added to three registers with migration of a view pixel. Drawing 12 (c) shows signs that the error is spread to each register, from the view pixel P02, when a view pixel moves to a pixel P02. When a view pixel is in a pixel P01 so that clearly if above-mentioned drawing 12 R> 2 (b) and above-mentioned drawing 12 (c) are compared, they are a register R2, a register R3, and a register R4. Although the error from a view pixel to a lower left pixel, a lower right pixel, and the pixel of right under is added to each register When a view pixel moves to a pixel P03 from a pixel P02, the error from a view pixel to a lower right pixel, the pixel of right under, and a lower left pixel will be added to each register, respectively.

[0110] Thus, it sets to the number transform processing of gradation of the 2nd example, and the error from a view pixel to a lower right pixel, the pixel of right under, and a lower left pixel is a register R2. Or register R4 Although added to three registers, whether each error is added to which register changes in order with migration of a view pixel. In this way, if the error is added in order, resetting each register to respectively suitable timing, moving a view pixel, the error from three continuous view pixels will be accumulated in each register in order so that it may explain below. The following and register R4 It takes for an example and explains concretely.

[0111] When a view pixel is in a pixel P00, as shown in drawing 12 R> 2 (a), error E00_RD from the view pixel P00 to a lower right pixel is added to a register R4. When adding error E00_RD so that it may mention later, it is a register R4. Since it is already reset, there is no error memorized. when a view pixel moves to a pixel P01, it is shown in drawing 12 (b) -- as -- register R4 **** -- error E01_D from the view pixel P01 to the pixel of right under It is added. register R4 **** -- since error E00_RD from the previous view pixel P00 is already added -- this error -- in addition, error E01_D from the view pixel P01 will be added. when

a view pixel moves to a pixel P02, it is shown in drawing 12 (c) -- as -- register R4 **** -- error E02_LD from the view pixel P02 to a lower left pixel is added. register R4 **** -- each error E00_RD and error E01_D from the previous view pixels P00 and P01 Since it is already added, in addition to these errors, error E02_LD from the view pixel P02 will be added. consequently, the time of adding the error from the view pixel P02 -- register R4 **** -- the error from three view pixels which the view pixel P00 thru/or the view pixel P02 follow will be accumulated. It sets to drawing 12 (c) and is a register R4. For the asterisk shown in the right, the error from the view pixel of the last in three continuous view pixels is a register R4. Being accumulated is shown. In this way, it is a register R4 about the error from three continuous pixels. If it finishes accumulating, the accumulated value will be written in an error buffer.

[0112] clear from above-mentioned explanation -- as -- register R4 **** -- since the error diffused in a pixel P11 is accumulated -- register R4 The accumulated value is written in the error buffer corresponding to a pixel P11. drawing 12 (c) -- setting -- register R4 from -- the arrow head of the void which faces to an error buffer -- register R4 Signs that the accumulated error is written in an error buffer are shown typically. Moreover, it is shown typically that the arrow head of the broken line of an error buffer and a pixel P11 to connect is an error buffer corresponding to a pixel P11. After writing the accumulated value in an error buffer, it is a register R4. A value is reset. the time of adding the error from the view pixel P00 in explanation of drawing 12 (a) -- register R4 already having been reset -- last view pixel P0-1 from -- the time of adding an error -- register R4 the error from three continuous pixels is accumulated -- having -- register R4 It is because it was reset.

[0113] Next, register R3 It pays its attention and explains. the time of adding the error from the view pixel P00, as shown in drawing 12 (a) -- register R3 **** -- three continuous view pixels P0 -2, P0-1, and the error from P00 are accumulated. Register R3 The asterisk shown in the right shows that the error from three continuous view pixels is accumulated, if the error from the view pixel P00 is accumulated. Then, after adding the error from the view pixel P00, it is a register R3. The accumulated value is written in an error buffer. Register R3 The accumulated error is pixel P1-1. The error accumulated since it was an error is pixel P1-1. It writes in a corresponding error buffer. In this way, when writing the accumulated error in an error buffer, it is a register R3. The value is reset.

[0114] here -- register R3 The timing reset and the above-mentioned register R4 if the timing reset is compared -- register R4 View pixel P0-1 before the error from the view pixel P00 is added from -- although it is reset after adding an error -- register R3 It is reset after adding the error from the view pixel P00. Therefore, register R3 Register R4 It receives, and it will be behind by one view pixel, and will be reset.

[0115] if a view pixel moves to a pixel P01 from a pixel P00 as shown in drawing 12 (b) -- register R3 **** -- the pixel P1-1 till then It replaces with and the new pixel P12 is assigned. since a pixel P12 is a pixel located in the lower right to the view pixel P01 -- register R3 **** -- error E01_RD diffused from a view pixel to a lower right pixel will be added. register R3 the time of the error from the view pixel P01 being added since the

value is reset previously -- register R3 **** -- there is no error currently added. when a view pixel moves to a pixel P02, it is shown in drawing 12 (c) -- as -- register R3 **** -- error E02_D diffused in the pixel of right under from a view pixel In addition to error E01_RD already added, it is added. furthermore, the time of a view pixel moving to the following pixel P03 -- register R3 **** -- the error from three view pixels which a pixel P01 thru/or a pixel P03 follow is accumulated -- ******** -- register R3 The accumulated value will be written in an error buffer. Register R4 mentioned above The accumulated error is a register R3, although it was written in the error buffer after adding the error from the view pixel P02. If it attaches, it is behind by one pixel, and after adding the error from the view pixel P03, it will be written in an error buffer.

[0116] Register R2 If it attaches, it is a register R3. It will receive and timing will be further overdue by one view pixel. Namely, register R3 Although the value accumulated in the register was written in the error buffer and the register was reset after adding the error from the view pixel P00 as shown in drawing 12 (a) if attached Register R2 If it attaches, after adding the error from the view pixel P01, the value accumulated in the register will be written in an error buffer, and a register will be reset.

[0117] Thus, register R2 Or register R4 The error from three view pixels which continue whenever a view pixel moves to three registers is accumulated in order, and is written in the error buffer one by one.

[0118] As mentioned above, as explained, it is the register R1 among four registers. The error used for decision of the dot formation existence of the following pixel is memorized. Register R2 Or every 1 pixel of three registers of a register R4 can be written in an error buffer from the pixel which are recording of the error for 3 pixels which continue whenever it will judge 1-pixel dot formation existence, if it is used in order to carry out sequential are recording of the gradation error ended.

[0119] C-2. The number transform processing of gradation of the 2nd example : drawing 1313 is a flow chart which shows the flow of the number transform processing of gradation of the 2nd example explained above. This processing as well as the number transform processing of gradation of the 1st example is performed by CPU102 of a computer 100. In addition, although the following explanation explains without specifying the color of ink, or the magnitude of a dot, processing same about the dot of every color and various magnitude is performed. Below, according to the flow chart of drawing 13 , the number transform processing of gradation of the 2nd example is explained focusing on difference with the number transform processing of gradation of the 1st example.

[0120] If the number transform processing of gradation of the 2nd example is started, it is the image data Cd about a view pixel. Diffusion error Ed It reads from RAM106 (step S400). Then, image data Cd Diffusion error Ed Register R1 By adding the error memorized, it is the amendment data Cx. It computes (step S402). Amendment data Cx When computing, it is a register R1. It resets (step S404). In this way, obtained amendment data Cx Predetermined threshold th Variable Cr which compares (step S406), will judge that a dot is formed if the amendment data are larger, and shows a decision result Value [meaning

forming a dot] "1" Write in (step S408). Otherwise, it judges that a dot is not formed and is Variable Cr. The value "0" meaning a dot not forming is written in (step S410).

[0121] In this way, if dot formation existence is judged, the gradation error E generated in connection with this will be computed (step S412). the gradation error E -- the 1st example -- the same -- amendment data Cx from -- it asks by subtracting a value (gradation value expressed by the view pixel forming a dot or by not forming a dot) as a result of a view pixel.

[0122] Subsequently, the multiplication of the gradation error E searched for at the predetermined error diffusion coefficient and predetermined step S412 which becomes settled for every pixel location by the error diffusion matrix is carried out, and the error searched for for every pixel location is added to each register. First, it is a register R1 about the error diffused in a pixel on the right from a view pixel. It adds (step S414). Subsequently, a register R2 thru/or a register R4 Each error is added to three registers. The flag is beforehand set to the register with which the error from the 3rd pixel of a continuous view pixel is added in these three registers, i.e., the register in which the asterisk was attached and shown in drawing 12 . In this way, the error diffused in the pixel at the lower left of a view pixel is added to the register to which the flag is set (step S416). As shown in the error diffusion matrix of drawing 6 (a), in the pixel at the lower left of a view pixel, it is the gradation error E and error diffusion coefficient K1-1. The value which carried out multiplication is distributed. The error which should be distributed to the pixel immediately under a view pixel is added to the register in front of [of the register with which the flag was set up among three registers adding an error] one (step S418). The error distributed to the pixel just under a view pixel carries out the multiplication of the gradation error E and the error diffusion coefficient K10, and is searched for. Here, for the register in front of [of the register with which the flag was set up] one, as shown in drawing 12 R> 2 (a), an asterisk is a register R3. When set up, as it is shown in a register R2 and drawing 12 (b), an asterisk is a register R2. When set up, it is a register R4. A register is pointed out. The value which carried out the multiplication of the error E which should be distributed to the pixel at the lower right of a view pixel, i.e., a gradation error, and the error diffusion coefficient K11 is added to the remaining registers in three registers (step S420).

[0123] In this way, if the value which carried out the multiplication of the predetermined error diffusion coefficient to the gradation error E is added to each register, the diffusion error accumulated in the register with which the flag is set will be written in the error buffer corresponding to the pixel at the lower left of a view pixel (step S422). For example, as shown in drawing 12 (c), when a view pixel is in a pixel P02, an error is written in the error buffer of the location corresponding to a pixel P11. register R4 with which the flag was set **** -- the error distributed from each pixel of a pixel P00, a pixel P01, and a pixel P02 is accumulated, and if it compares with the usual error diffusion method shown in drawing 5 , it turns out that equivalent processing is substantially performed also in processing of the 2nd example. In this way, if the accumulated error is written in an error buffer, the register will be reset in order to accumulate the error newly distributed (step

S424).

[0124] If the processing about one view pixel is ended as mentioned above, after moving a flag to the register in front of [of a current register] one (step S426), a series of processings which continue until it will return to step S400 and the processing about all pixels will be completed, if it judged whether processing was ended about all pixels (step S428) and the unsettled pixel remains are repeated. If the processing about all pixels is completed, it will escape from the number transform processing of gradation of the 2nd example, and will return to image data-conversion processing of drawing 4 .

[0125] In addition, it sets to step S414 and is a register R1 about an error. Instead of adding, it is a register R1. It is good also as overwriting. If it carries out like this, it will set to step S404, and it is a register R1. Since the processing to reset is omissible, it is desirable. If it changes into the processing which similarly overwrites the processing which adds an error to the remaining registers in step S420, since it becomes possible to omit the processing which resets a register in step S424, it is desirable.

[0126] As mentioned above, what is necessary is just to write the diffusion error of one pixel in an error buffer in the number transform processing of gradation of the 2nd explained example, whenever it judges the dot formation existence of one pixel. Therefore, the time amount for diffusing an error in an error buffer can be greatly shortened to the usual error diffusion method. As mentioned above, when the usual error diffusion method is used for the number transform processing of gradation, the time amount which diffusion with error takes has accounted for the comparatively big rate in the time amount which the number transform processing of gradation takes. It becomes possible from this to shorten the processing time, if the number transform processing of gradation of the 2nd example is adopted, as a result to print an image quickly.

[0127] In addition, even when the number conversion approach of gradation of the 1st above-mentioned example is used, the time amount required since an error is diffused can be shortened, but if the approach of the 2nd example is adopted, a diffusion time with error can be shortened still more efficiently. That is, although per dot formation decision was sufficient in comparison with the usual error diffusion method only by adding an error to an error buffer by the frequency for 2.5 pixels when the approach of the 1st above-mentioned example was used, if the approach of the 2nd example is used, the frequency which writes an error in per dot formation decision can be decreased even to 1 pixel. Moreover, in the case of the example [1st] shown in drawing 7 , although six registers were used as a middle buffer, it enables only the part to only use four registers in the case of the approach of the 2nd example, and to use the register of CPU102 for other purposes.

[0128] C-3. Modification : although the error generated in the view pixel is accumulated in four registers in the number transform processing of gradation of the 2nd example mentioned above, it is the register R1 of these. Other three registers R2 Or register R4 How to use a register differs a little. Namely, register R1 If attached, the error which should be distributed to the pixel on the right of a view pixel was always overwritten, but

about other three registers, the relative position to a view pixel was not fixed, and whenever the view pixel moved, it was moved.

[0129] On the other hand, it is good also as fixing to drawing 14 the pixel location where each register corresponds to a view pixel so that it may be shown. namely, register R1 **** -- the error which should be distributed to the pixel on the right of a view pixel always overwrites -- having -- register R2 **** -- the error which should be distributed to the pixel at the lower right of a view pixel -- register R3 **** -- the error which should be distributed to the pixel just under a view pixel -- register R4 ****. -- the pixel which should be distributed to the pixel at the lower left of a view pixel may always be made to be added. Thus, the error accumulated in each register to compensate for migration of a view pixel so that it may mention later after fixing the physical relationship of each register and a view pixel is moved to the register used as sequential. If it carries out like this, the processing which can write the error accumulated in the always same register in an error buffer, and writes a diffusion error in an error buffer from a register since ** is good can be simplified. Moreover, since the error diffusion coefficient used with each register also serves as the always same value, the processing which computes the error added to each register also becomes possible [simplifying] from the gradation error generated in the view pixel.

[0130] Hereafter, the number transform processing of gradation as a modification of such 2nd example is explained briefly, referring to drawing 15 . Drawing 15 (a) shows the condition of having judged the dot formation existence about the view pixel P00. Four rectangles shown in the right-hand side of drawing show each register typically. register R1 **** -- the error which should always be distributed to the pixel on the right-hand side of a view pixel is overwritten. moreover, register R2 **** -- the error which should always be distributed to the pixel at the lower right of a view pixel adds -- having -- register R3 **** -- the error which should be distributed to the pixel just under a view pixel -- register R4 **** -- the error which should be distributed to the pixel at the lower left of a view pixel is added.

[0131] Thus, since the error which should be distributed to the pixel of the always same location to a view pixel is added to each register, each error diffusion coefficient serves as the always same value. Namely, register R1 An error diffusion coefficient is always the error diffusion coefficient K01 to the pixel on the right-hand side of a view pixel, and is a register R2. An error diffusion coefficient is always the error diffusion coefficient K11 to the pixel at the lower right of a view pixel (refer to drawing 6). Similarly, it is a register R3. An error diffusion coefficient is always K10, and is a register R4. An error diffusion coefficient is always K1-1. It becomes. In this way, since the error diffusion coefficient of each register serves as the always same value, the error which should be added to each register is easily computable from the gradation error generated in the view pixel.

[0132] When adding the gradation error E00 generated in the view pixel to each register, it is a register R4. The value accumulated in each register is shifted to the register set to one at a time at the same time it performs processing which writes the value of the error accumulated in an error buffer. Namely, register R3 It is a register R4 about a value. It is

made to move and is a register R2. It is a register R3 about a value. It is made to move (see the lower part of drawing 15 (a)).

[0133] If the above processing is ended, subsequently to a pixel P01, a view pixel will be moved and same processing will be performed again. When adding the gradation error E01 generated in the view pixel P01 to each register, applying a predetermined error diffusion coefficient, it is a register R4. Processing which writes the value accumulated in an error buffer is performed. the time of the processing which adds an error to each register as shown in drawing 15 (b) being completed since processing which shifts previously every one value memorized by each register was performed -- register R4 **** -- the error from three pixels will always be accumulated. Therefore, what is necessary is just to write the error accumulated in the always same register in an error buffer in the modification of the 2nd example. For this reason, the processing which writes a diffusion error in an error buffer from a register can be simplified. In addition, although the processing to which the error accumulated in each register is shifted to the next register is newly added in the processing shown in drawing 15 , since processing to which a value is moved between such registers can be performed very quickly, the increment in the processing time by that is small.

[0134] In the number transform processing of gradation of the 2nd example mentioned above, or the number transform processing of gradation of the modification of the 2nd example, in order to avoid that explanation becomes complicated, each explained the error diffusion matrix as what uses the matrix of the narrowest drawing 6 (a) of the diffusion range of with error. But as for the error diffusion matrix to be used, it is needless to say that other matrices may be used, without being limited to the matrix of drawing 6 (a). Moreover, of course, same processing may be substantially performed by utilizing cache memory like the case in the 1st example.

[0135] Moreover, the in-between diffusion error which should be diffused in a surrounding pixel is not added to a register, but the gradation error itself is memorized to the middle buffer, and you may make it compute the final diffusion error which should be written in an error buffer whenever it memorizes a new gradation error also in the number conversion approach of gradation of the 2nd example. Although this approach has the difference in whether it is the diffusion error which the value memorized by the register is accumulating, or it is a gradation error, since the error written in an error buffer serves as the same value, equivalent processing can be performed substantially.

[0136] D. The 3rd example : although each example explained above explains as what diffuses an error using the always same error diffusion matrix, in the actual number transform processing of gradation, it may be used from the request on image quality, changing a narrow error diffusion matrix of the diffusion range like drawing 6 (a), and the large matrix of diffusion range like drawing 6 (c).

[0137] That is, in an error diffusion method, in order for a dot to prevent being formed by the specific periodic pattern, it may be used, changing two or more kinds of error diffusion matrices at random. Moreover, when the gradation value of image data is judged to be

smaller than a predetermined, threshold small enough, and to form a dot by JP,7-226841,A, the technique in which dot density improves the dispersibility of a dot in a **** field is indicated by it by using the large error diffusion matrix of the diffusion range, and diffusing an error. Although explanation is omitted about the reason the dispersibility of a dot is improvable by carrying out like this Here according to the technique indicated by JP,7-226841,A, the gradation value of image data is smaller than the predetermined, threshold thm small enough. And when it is judged that a dot is formed, the large matrix of the error diffusion range shown in drawing 6 (c) is used, and in being other, it shall use the narrow matrix of the error diffusion range shown in drawing 6 (a). In such a case, if the approach of the 3rd example explained below is used, it will become possible to shorten effectively the time amount which the number transform processing of gradation takes.

[0138] Drawing 16 is the flow chart which showed the flow of the processing which performs the number transform processing of gradation, changing two error diffusion matrices. Hereafter, according to the flow chart of drawing 16 , the number transform processing of gradation of the 3rd example is explained. In addition, although it explains without distinguishing the color of a dot, and the magnitude of a dot in order to avoid that explanation makes it complicated as well as the number transform processing of gradation of each example mentioned above, the following processings are performed for every dot of every color ink and various magnitude.

[0139] If the number transform processing of gradation of the 3rd example is started, it is the image data Cd of a view pixel first. Diffusion error Ed It reads (step S500). Image data Cd Diffusion error Ed RAM106 memorizes. Register R1 The error and image data Cd from the pixel of the left-hand memorized And diffusion error Ed It adds and is the amendment data Cx. It computes (step S502). Amendment data Cx for which it asked The predetermined threshold th is compared (step S504). Amendment data Cx It is judged that a dot will be formed in a view pixel if the direction is large (step S504: yes). Variable Cr showing the result of dot formation decision After writing in the value "1" meaning forming a dot (step S506), the gradation error produced in a view pixel is computed by having formed the dot (step S508). Amendment data Cx It is Variable Cr if smaller than the predetermined threshold th (step S504: no). The gradation error produced in a view pixel by that is computed by writing in the value "0" meaning not forming a dot (step S510) (step S512).

[0140] When the dot is formed in a view pixel, it is the image data Cd of a view pixel. The predetermined threshold thm for changing an error diffusion matrix is compared (step S514). Image data Cd Since it is thought that the dot was formed in the field of (step S514:no) and small image data by chance when smaller than a threshold thm, each error diffusion coefficient is set up according to the large error diffusion matrix (matrix shown in drawing 6 (c) here) of the error diffusion range (step S516). that is, it is shown in drawing 6 (c) -- as -- the error diffusion coefficient K10 of the pixel just under a view pixel -- one fourth -- setting up -- each error diffusion coefficient K01 and K11 of the pixel on the right of a view pixel, a lower right pixel, and a lower left pixel, and K1-1 **** -- one eighth is set up.

the error diffusion coefficients K02, K03, K12, K13, and K of the pixel [pixel / view] which exists in the distance further -- 1-2 and K1-3 **** -- 1/16 is set up.

[0141] Subsequently, about six pixels which are far away from a view pixel, the value which carried out the multiplication of a gradation error and the error diffusion coefficient of each pixel is directly added to each error buffer (step S518). This is explained using drawing 17 . The condition that drawing 17 judged dot formation existence about the view pixel P00, consequently the gradation error E00 occurred is shown. This gradation error E00 is diffused in the large range using the error diffusion matrix shown in drawing 6 (c). That is, it is the pixel P02 whose gradation error E00 of a pixel P00 is in the outside of a broken line although four pixels enclosed with a broken line usually thick in drawing 17 are made to diffuse an error, pixel P03, pixel P12, pixel P13, pixel P1-2, and pixel P1-3. Six pixels are also made to diffuse an error. In processing of step S518, the value which carried out the multiplication of the gradation error E00 and the error diffusion coefficient of each pixel about these six pixels is directly added to the error buffer of each pixel.

[0142] If an error is added to the error buffer of a distant pixel, an error will be diffused in the register of each pixel which is inside a broken line (step S522). That is, in the number transform processing of gradation of the 3rd example, although a direct error is added to an error buffer about the pixel of the distant place of a view pixel, about each pixel of the view pixel circumference, an error is diffused like the 1st above-mentioned example or the 2nd example using a middle buffer. Below, when diffusing an error according to the approach of the modification of the 2nd example, it explains taking the case of the case where the error to the pixel which is in the always same physical relationship to a view pixel is added to each register.

[0143] First, it is a register R1 at the value which carried out the multiplication of the right-hand side error diffusion coefficient K01 and the right-hand side gradation error E00 to a pixel of a view pixel. A value is updated. Next, it is a register R2 about the value which carried out the multiplication of the error diffusion coefficient K11 and the gradation error E00. It is a register R3 about the value which added and carried out the multiplication of the error diffusion coefficient K01 and the gradation error E00. It adds and is error diffusion coefficient K1-1. It is a register R4 about the value which carried out the multiplication of the gradation error E00. It adds (see drawing 15 (a)). The above processing is performed at step S522. Then, register R4 Processing which adds the value accumulated to the corresponding error buffer of a pixel is performed (step S524). Since a view pixel is a pixel P00 here as shown in drawing 17 , it is pixel P1-1. It is a register R4 to an error buffer. What is necessary is just to add a value. When the processing added to an error buffer is completed, it is a register R3. It is a register R4 about the value accumulated. Register R2 It is a register R3 about the value accumulated. It is made to shift (step S526). Register R2 after performing processing which shifts the value of each register The value is reset (step S528).

[0144] On the other hand, even if it forms the dot in a view pixel when the dot is not formed in a view pixel (step S504: no) or, it is image data Cd. In being larger than the

predetermined threshold thm (step S514: yes), according to a matrix with the narrower error diffusion range (matrix shown in drawing 6 R> 6 (a) here), it sets up each error diffusion coefficient (step S518). That is, according to the error diffusion matrix of drawing 6 (a), one fourth is set as the error diffusion coefficient of four pixels of the view pixel circumference. In this way, it is a register R4 by performing processing of step S522 thru/or step S528 mentioned above using the set-up error diffusion coefficient. The accumulated error is added to an error buffer.

[0145] If the dot formation existence of a view pixel is judged as mentioned above and the diffusion error about 1 pixel is written in an error buffer, it will judge whether processing was ended about all pixels (step S530). When the unsettled pixel remains, it returns to step S500 again, and the image data about a new view pixel and a diffusion error are read from RAM106, and they are these values and registers R1. The error memorized is used and it is the amendment data Cx. It computes (step S502). Henceforth, if a series of above processings are repeated and processing is ended about all pixels until an unsettled pixel is lost, it will return to the image data-conversion processing which escapes from the number transform processing of gradation of the 3rd example, and is shown in drawing 4 .

[0146] If the number transform processing of gradation is performed using the approach of the 3rd example mentioned above, when performing the number transform processing of gradation, changing the large error diffusion matrix of the diffusion range, and the narrow error diffusion matrix of the diffusion range, the processing time can be shortened efficiently. That is, many registers are needed, when using the large error diffusion matrix of the diffusion range shown in drawing 6 (c) and it is going to accumulate all errors in a register. If it explains with reference to drawing 17 , the register for 10 pixels which added the register for 4 pixels which are inside the range enclosed with a broken line, and the register for 6 pixels of the outside of a broken line will be used. in order to add an error to these registers, carry out using the approach of the 2nd example which uses a flag -- carry out using the approach of the modification of the 2nd example to which the value of a register is shifted -- if the number of registers increases too much, time amount will be needed for actuation of a register, and the processing time as the whole will be made to increase

[0147] On the other hand, since the number transform processing of gradation can be performed without making the number of the registers adding an error increase even when using the approach of the 3rd example mentioned above and the matrix which diffuses an error is used for the even if very large range, processing is simplified as a whole and it becomes possible to avoid that the processing time increases.

[0148] It is used when it is judged that the large error diffusion matrix of the diffusion range has the small gradation value of image data enough, and a dot is formed, as especially mentioned above. If the gradation value of image data is fully small, since the probability judged to form a dot is small, it can be said that the large error diffusion matrix of the diffusion range is a matrix with low operating frequency. Therefore, when an error must be diffused in the range large very rarely, the number transform processing of

gradation can be quickly performed as a whole by supposing that a direct error is added to an error buffer, even if it takes time amount somewhat, utilizing a middle buffer in a case usual [without the need of diffusing so large the range], and diffusing an error quickly. Moreover, when spread in the large range, adding a direct error to the error buffer of a distant pixel, then the number of registers used so much as a middle buffer can be saved, and it becomes possible to increase the efficiency of processing further by using the register which floated for other processings.

[0149] In addition, in the 3rd above-mentioned example, only when it supposes that an error diffusion matrix is switched and the large error diffusion matrix of the diffusion range is chosen, the direct error of a distant pixel shall be diffused, but when switching a matrix, it is not necessarily limited. For example, when the error diffusion matrix to be used is a large matrix of the diffusion range, you may make it diffuse an error collectively based on the gradation error of the number of predetermined pixels by making a distant pixel diffuse and memorize a direct error, and utilizing a middle buffer for a nearby pixel. If it carries out like this, it will become possible collectively to save a middle buffer required in order to make an error diffuse and memorize.

[0150] Moreover, in the 3rd example explained above, by utilizing a middle buffer, in order to summarize a diffusion error to a circumference pixel and to make it spread and memorize from the gradation error of the number of predetermined pixels, it cannot be overemphasized that various kinds of approaches explained as the 1st example thru/or the 2nd example can be suitably applied to an approach.

[0151] E. The 4th example : finally the approach of the various examples mentioned above has distributed the gradation error to the error buffer of a circumference pixel. From this semantics, technique similar to the approach called the so-called error diffusion method can be considered. Of course, also when memorizing the gradation error generated by decision of dot formation existence to the view pixel like the approach called the so-called average error minimum method, reading a gradation error from a circumference pixel on the occasion of decision of the dot formation existence of a non-judged pixel and making a dot formation judgment, it is possible to shorten the time amount which dot formation decision takes by utilizing a middle buffer. Below, the 4th example which performs the number transform processing of gradation using such an approach is explained.

[0152] E-1. The principle which shortens the time amount of the number transform processing of gradation in the 4th example : drawing 18 is the explanatory view showing the principle which shortens the time amount which decision of dot formation existence takes by utilizing a middle buffer in the approach of the 4th example. Drawing 18 (a) shows signs that the formation existence of a dot is judged about the view pixel P00. As preparation explaining the approach of the 4th example, drawing 18 (a) is diverted and the approach called the so-called average error minimum method is explained briefly.

[0153] By the average error minimum method, the gradation error produced by decision of dot formation existence is memorized to the error buffer corresponding to the pixel. if this is **(ed) and explained to drawing 18 (a), it is in the chart on the left with "E0-1" -- pixel

P0-1 the generated gradation error -- being shown -- **** -- pixel P0-1 within the limit -- E0-1 ** --- by displaying Gradation error E0-1 Pixel P0-1 What is memorized by the error buffer is shown typically. Moreover, it is shown that the slash's being attached all over drawing is dot formation decision ending. By the average error minimum method, each gradation error is memorized by the pixel [finishing / decision of dot formation existence] as illustrated. In judging dot formation existence about the non-judged view pixel P00, each gradation error is read from a pixel [finishing / decision of the circumference], and it judges the dot formation existence of the view pixel P00, taking these errors into consideration. More, according to the relative position to the view pixel of a circumference pixel, the predetermined weighting factor is beforehand set to the detail as some are illustrated by drawing 19 , it is the value which applied the predetermined weighting factor to the error read from each surrounding pixel, and amendment data are computed by amending the image data of a view pixel. In addition, in drawing 19 , the pixel to which the slash is given is a view pixel, and the numeric value currently displayed on each pixel is the weighting factor set as the pixel. In this way, by comparing with a predetermined threshold the amendment data for which it asked, the dot formation existence about a view pixel is judged. If dot formation existence is judged about a view pixel, the gradation error produced in that will be computed and the computed gradation error will be memorized to the error buffer of a view pixel. By the average error minimum method, formation existence of a dot is judged for every pixel by repeating the above processings.

[0154] The gradation error of the pixel of plurality whenever it makes a judgment about one pixel in order to judge dot formation existence using the average error minimum method as mentioned above must be read from an error buffer, and in order that dot formation existence may judge, a certain amount of time amount is needed.

[0155] On the other hand, in the number transform processing of gradation of the 4th example explained below, though processing equivalent to the above-mentioned average-error minimum method is performed mathematically, it is possible by utilizing a middle buffer to judge the existence of dot formation quickly. Hereafter, by contrasting with the above-mentioned average error minimum method explains the principle which shortens the processing time in the number transform processing of gradation of the 4th example, referring to drawing 18 . In addition, in order to avoid complicated-ization of explanation, below according to a setup of the weighting factor shown in drawing 19 (a), the gradation error of a circumference pixel shall be taken into consideration.

[0156] Drawing 18 (a) is the explanatory view showing signs that the dot formation existence about the view pixel P00 is judged in the number transform processing of gradation of the 4th example. The gradation error E-1-1 generated in the pixel P-1-1 like the average error minimum method mentioned above in decision of the dot formation existence about the view pixel P00, and pixel P-10 Generated gradation error E-10 Pixel P-11 Generated gradation error E-11 Pixel P0-1 Gradation error E0-1 generated It is used. Four rectangles on the right-hand side of drawing 18 (a) show typically four registers used as a middle buffer. Here [the facilities of explanation to], it is R1, R2, R3, and R4 to each

register. Agreement will be attached and distinguished, respectively.

[0157] The gradation error of the pixel which has a position relation to a view pixel is overwritten by each register. namely, register R1 **** -- the gradation error in the pixel which is always in the upper right of a view pixel memorizes -- having -- register R2 **** -- the gradation error in the pixel which exists right above a view pixel memorizes -- having -- register R3 **** -- the gradation error of the pixel at the upper left of a view pixel -- register R4 **** -- the gradation error of a pixel on the left is memorized. About the processing which memorizes each gradation error to each register, it mentions later.

[0158] In the number transform processing of gradation of the 4th example, the amendment data in a view pixel are computed in consideration of the gradation error in each pixel memorized by these four registers, and the weighting factor beforehand set up for every pixel. That is, by the average-error minimum method mentioned above, the places beforehand memorized by each register here differ greatly to reading the gradation error in each pixel from an error buffer. In this way, by comparing with a predetermined threshold the amendment data for which it asked, the dot formation existence about a view pixel is judged. If dot formation existence is judged, the gradation error E00 in the view pixel P00 will be computed continuously. A gradation error can be searched for by taking a difference with a value as a result of amendment data and a view pixel.

[0159] In this way, if the gradation error E00 about a view pixel is searched for, in order to make a dot formation judgment about a new view pixel, the actuation shown in drawing 18 (b) to each register is added. First, register R4 The error memorized is written in an error buffer. it explained using drawing 18 (a) -- as -- register R4 **** -- since the gradation error always produced in the pixel on the left of a view pixel is memorized -- register R4 A value overwrites the error buffer of the pixel on the left of a view pixel. Subsequently, it is a register R4 about the gradation error E00 about the pixel P00 for which it asked previously. It overwrites and is a register R2 further. It is a register R3 about a value. Register R1 It is a register R2 about a value. It is made to move, respectively. That is, the value of each register is moved corresponding to a view pixel moving to the pixel P01 on the right from a pixel P00. these processings move data between the registers of the CPU102 interior -- being sufficient -- since -- it can carry out very quickly. At the end, the gradation error in the pixel at the upper right of a new view pixel is read from an error buffer, and it is a register R1. It is made to memorize.

[0160] By adding the above actuation, the value memorized by each register will be in the condition of drawing 18 (c) from the condition shown in the right-hand side of drawing 18 (a). The value memorized by each register of drawing 18 (c) so that clearly, if drawing 18 (a) is compared with drawing 18 (c) is the value of each register when changing a view pixel into a pixel P01 from a pixel P00 in drawing 18 R> 8 (a). Therefore, if the above processings are repeated and are performed, a dot formation judgment of a new pixel can be made one after another. Thus, by the approach of the 4th example, it becomes possible by storing in the register to make the frequency which reads a gradation error from an error buffer mitigate greatly of the gradation error used continuously.

[0161] E-2. The number transform processing of gradation of the 4th example : explain briefly that the processing for actually performing hereafter the number transform processing of gradation of the 4th example mentioned above flows. Drawing 20 is the flow chart which showed the flow of the number transform processing of gradation of the 4th example. In addition, although it explains below like the number transform processing of gradation of the various examples mentioned above, without specifying the class of ink, and the magnitude of a dot in order to avoid complicated-ization of explanation, same processing is performed for every dot of every color and various magnitude.

[0162] If the number transform processing of gradation of the 4th example is started, it is the image data Cd of a view pixel first. It reads (step S600), and the gradation error of the pixel at the upper right of a view pixel is continuously read from an error buffer, and it is a register R1. It memorizes (step S602). Consequently, as explained using drawing 18 , the gradation error in each pixel of the view pixel circumference is memorized by each register.

[0163] Subsequently, the multiplication of the error memorized by each register and the predetermined weighting factor is carried out for every register, and it is the amendment data Cx of a view pixel. It computes (step S604). The ****** value acquired by carrying out the multiplication of the error remembered to be a weighting factor predetermined [these] since the predetermined weighting factor is set to the pixel of the view pixel circumference for every pixel as shown in drawing 19 for every register, and image data Cd which read previously By adding, it is the amendment data Cx of a view pixel. It computes. In this way, amendment data Cx for which it asked The predetermined threshold th is compared (step S606). Amendment data Cx The value "0" meaning judging that a dot is formed in a view pixel, if the direction is large, writing the value "1" meaning forming a dot in the variable Cr which shows a decision result (step S608), otherwise, not forming a dot is written in (step S610). Then, the gradation error produced in a view pixel by this decision result is computed (step S612). A gradation error is searched for by subtracting a value from the amendment data of a view pixel a result like the various examples mentioned above.

[0164] If the gradation error in a view pixel is searched for as mentioned above, as explained using drawing 18 (b), a series of following actuation will be added to each register. First, register R4 The gradation error memorized is written in an error buffer (step S614). Next, it is a register R4 about the gradation error searched for at step S612. It writes in (step S616). Then, register R2 It is a register R3 about a value. It moves (step S618) and is a register R1. It is a register R2 about a value. It is made to move (step S620).

[0165] A series of continuing processings are repeated until it will return to step S600 and processing will be completed about all pixels, if it judged whether dot formation decision was ended about all pixels when actuation of the above registers was completed (step S622) and the non-judged pixel remains. If the dot formation existence of all pixels is judged, it will escape from the number transform processing of gradation of the 4th example, and will return to the image data-conversion processing shown in drawing 4 .

[0166] What is necessary is just to read the gradation error for 1 pixel from an error buffer in the number transform processing of gradation of the 4th example explained above,

whenever it judges 1-pixel dot formation existence. When applying the average-error minimum method, as compared with reading the diffusion error for 4 pixels from an error buffer, whenever it makes a 1-pixel judgment, as mentioned above, the frequency where data are written to an error buffer can be reduced, and the part and the number transform processing of gradation can be performed quickly. Of course, the approach and the average error minimum method of the 4th example are performing processing equivalent mathematical completely, and if the approach of the 4th example is used, they can obtain a high-definition image like the case where the average error minimum method is used.

[0167] In addition, in the 4th example mentioned above, it explained as that to which the value illustrated to drawing 19 (a) is set as a setup of a weighting factor. It cannot be overemphasized that it can consider as various setup according to the request of image quality without limiting an actual weighting factor to a setup illustrated to drawing 19 R> 9. Moreover, although four registers were used as a middle buffer in the 4th above-mentioned example corresponding to using a setup illustrated to drawing 19 (a), of course according to a setup of the weighting factor to be used, more registers may be needed.

[0168] Furthermore, in the 4th example mentioned above, although explained as a middle buffer as a thing using the register built in CPU102, of course, the storage element which can be written at high speeds, such as not only a register but cache memory, may be used.

[0169] As mentioned above, although various kinds of examples have been explained, this invention is not restricted to the example of all above, and can be carried out in various modes in the range which does not deviate from the summary.

[0170] For example, in various above-mentioned examples, it explained as what judges the formation existence of a dot based on the size relation between the amendment data of each pixel, and a predetermined threshold. Of course, the approach of judging the existence of dot formation can apply the various approaches of not only the approach mentioned above but common knowledge.

[0171] Moreover, in various above-mentioned examples, in order to avoid complicated-ization of explanation, the class of dot formed was explained to each pixel as one kind as what is formed and cannot take only two conditions, or [whether a dot is formed or / that there is nothing]. Of course, it is good also as formation of two or more kinds of dots from which the magnitude of a dot or ink concentration differs being possible. For example, formation of two kinds of dots of size and smallness may be enabled, and the formation existence of a dot may be judged as follows. That is, two thresholds th1 and th2 (however, referred to as th1>th2) are set up, if the amendment data of a pixel are larger than a threshold th1, it will judge that a large dot is formed, if larger smaller than a threshold th1 than a threshold th2, it will judge that a small dot is formed, and when smaller than a threshold th2, it is judged that a dot is not formed. The gradation error generated in each pixel is computable by subtracting a value from the amendment data of the pixel a result.

[0172] In addition, in the above various examples, although an error buffer will be logically

prepared for every pixel, it prepares only the error buffer for a number raster in practice, and the error buffer which is the pixel dot formation existence was judged to be is diverted to some other purpose as an error buffer of other pixels. Although it was the purpose which avoids complicated-ization of explanation, and it explained in the various above-mentioned examples as if the error buffer for all pixels was prepared, of course, the error buffer for a number raster may be repeated and used.

[0173] Moreover, the software program (application program) which realizes an above-mentioned function may be supplied and performed to the main memory or external storage of a computer system through a communication line. Of course, the software program memorized by CD-ROM and the flexible disk may be read and performed.

[0174] Moreover, although the various examples mentioned above explained the image data-conversion processing including the number transform processing of gradation as what is performed within a computer, a part or all of image data-conversion processing may be performed a printer side using the image processing system of dedication.

[0175] Furthermore, an image display device may be a liquid crystal display with which gradation expresses the image which changes continuously by necessarily not being limited to the airline printer which forms an ink dot on print media and prints an image, and distributing the luminescent spot by the suitable consistency on a liquid crystal display screen for example.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the outline block diagram of the printing system of this example.

[Drawing 2] It is the explanatory view showing the configuration of the computer as an image processing system of this example.

[Drawing 3] It is the outline block diagram of the printer as an image display device of this example.

[Drawing 4] It is the flow chart which shows the flow of the image data-conversion processing performed with the image processing system of this example.

[Drawing 5] It is the explanatory view showing notionally signs that the formation existence of a dot is judged using an error diffusion method.

[Drawing 6] It is the explanatory view which illustrates signs that the error diffusion coefficient is set up for every pixel.

[Drawing 7] It is the explanatory view showing the principle which shortens the processing time in the number transform processing of gradation of the 1st example.

[Drawing 8] It is the flow chart which shows the flow of the number transform processing of gradation of the 1st example.

[Drawing 9] It is the explanatory view showing the case where the error of many pixels is diffused at a time in the number transform processing of gradation of the 1st example.

[Drawing 10] It is the flow chart which shows the flow of the number transform processing

of gradation of the modification of the 1st example.
[Drawing 11] It is the explanatory view showing the principle which shortens the processing time in the number transform processing of gradation of the 2nd example.
[Drawing 12] It is the explanatory view showing the outline of processing of writing data with a middle buffer and an error buffer in the number transform processing of gradation of the 2nd example.
[Drawing 13] It is the flow chart which shows the flow of the number transform processing of gradation of the 2nd example.
[Drawing 14] To a view pixel when [the pixel location where each register corresponds] fixed, it is the explanatory view which illustrates signs that each register and the pixel of the view pixel circumference correspond.
[Drawing 15] It is the explanatory view showing the principle which shortens the processing time in the number transform processing of gradation of the modification of the 2nd example.
[Drawing 16] It is the flow chart which shows the flow of the number transform processing of gradation of the 3rd example.
[Drawing 17] It is with the case where the large matrix of the diffusion range is used in the number transform processing of gradation of the 3rd example, and the case where the narrow matrix of the diffusion range is used, and is the explanatory view comparing and showing the range which a gradation error diffuses.
[Drawing 18] It is the explanatory view showing the principle which shortens the processing time in the number transform processing of gradation of the modification of the 4th example.
[Drawing 19] In the number transform processing of gradation of the modification of the 4th example, it is the explanatory view which illustrates signs that the weighting factor is set up for every pixel.
[Drawing 20] It is the flow chart which shows the flow of the number transform processing of gradation of the 4th example.
[Description of Notations]
10 -- Computer
12 -- Printer driver
20 -- Color printer
100 -- Computer
102 -- CPU
104 -- ROM
106 -- RAM
108 -- Peripheral-device interface P-I/F
109 -- Disk controller DDC
110 -- Network Interface Card NIC
112 -- Video interface V-I/F
114 -- CRT

116 -- Bus
118 -- Hard disk
120 -- Digital camera
122 -- Color scanner
124 -- Flexible disk
126 -- Compact disk
200 -- Color printer
230 -- Carriage motor
235 -- Paper feed motor
236 -- Platen
240 -- Carriage
241 -- Print head
242,243 -- Ink cartridge
244 -- Head for ink regurgitation
260 -- Control circuit
261 -- CPU
262 -- ROM
263 -- RAM
300 -- Communication line
310 -- Storage